

0- 792857

сч

На правах рукописи

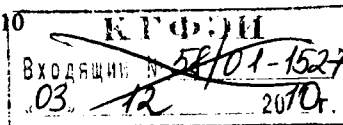
Щербаков Сергей Михайлович

**ЭКОНОМИКО-МАТЕМАТИЧЕСКИЕ МОДЕЛИ ПРОЦЕССОВ
ИСПОЛЬЗОВАНИЯ ИНТЕРНЕТ-ПРИЛОЖЕНИЙ:
МЕТОДОЛОГИЯ ПОСТРОЕНИЯ И
ИНСТРУМЕНТАРИЙ РАЗРАБОТКИ**

Специальность 08.00.13 – Математические и инструментальные
методы экономики

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
доктора экономических наук

Ростов-на-Дону – 2010



Работа выполнена в ГОУ ВПО «Ростовский государственный экономический университет (РИНХ)»

Научный консультант: доктор экономических наук, профессор
Хубаев Георгий Николаевич

Официальные оппоненты: доктор экономических наук, профессор
Емельянов Александр Анатольевич

доктор экономических наук, профессор
Матвеева Людмила Григорьевна

доктор экономических наук, профессор
Попова Елена Витальевна

Ведущая организация: ГОУ ВПО «Московский государственный
университет экономики, статистики и
информатики (МЭСИ)»

Защита диссертации состоится 20 декабря 2010 г. в 11⁰⁰ часов на заседании диссертационного совета ДМ 212.209.03 в Ростовском государственном экономическом университете (РИНХ) по адресу: 344002, г. Ростов-на-Дону, ул. Б. Садовая, 69, ауд. 231.

С диссертацией можно ознакомиться в научной библиотеке Ростовского государственного экономического университета (РИНХ).

Автореферат разослан «19» ноября 2010 г.



Ученый секретарь
диссертационного совета

И.Ю. Шполянская

Актуальность темы диссертационного исследования. Всемирная сеть Интернет оказывает влияние на различные стороны жизни современного общества. Все большее число предприятий и организаций обращается в своей деятельности к возможностям сети Интернет и интернет-приложений - информационные и коммуникационные средства используются для взаимодействия с клиентами, партнерами и филиалами, для расширения географии бизнеса, для рекламы и продвижения товаров, для модернизации бизнес-процессов (деловых процессов) организации. Активно распространяются и такие формы бизнеса, как электронная коммерция, продажа контента, баннерной и контекстной рекламы. Развитие принципов, технологий, программных и аппаратных средств, созданных для сети Интернет, определило возможность их использования и для совершенствования корпоративных информационных систем.

Под интернет-приложениями понимается класс информационных систем, ориентированных на использование технологий и стандартов сети Интернет.

Использование интернет-приложений позволяет сократить затраты труда на исполнение бизнес-процессов, повысить объемы продаж и выручки, способствует достижению иных целей организации. С другой стороны, разработка и сопровождение интернет-приложений требует существенных трудовых и финансовых затрат.

Построение и использование интернет-приложений требует принятия значительного числа технических и управленческих решений. Задача осложняется тем, что количественное и качественное развитие современных интернет-приложений привело к необходимости рассматривать их как сложные информационные системы, которые могут включать тысячи пользователей и содержать сотни тысяч элементов контента, поддерживать динамические возможности, предоставлять коммуникационные и вычислительные сервисы, взаимодействовать с другими информационными системами. Множество существующих технологий позволяет разрабатывать интернет-приложения различного масштаба и функциональности. От совокупности принятых решений зависит качество системы и величина затрат на ее разработку и эксплуатацию.

Таким образом, очевидна актуальность изучения экономической стороны процессов сравнения, выбора и использования интернет-приложений в деятельности предприятий и организаций. При этом сложность интернет-приложений определяет необходимость обращения к методам экономико-математического моделирования.

Степень изученности проблемы. Экономические аспекты сети Интернет и интернет-приложений рассматриваются в работах отечественных и зарубежных ученых: Э. Бриньолфсона (E. Brynjolfsson), Дж. Бэйли (J.P. Bailey), Х.Р. Вэриана (Hal R. Varian), С.А. Дятлова, К. Клаффи (Kc Claffy), А.Б. Курицкого, Л. МакКнайта (Lee W. McKnight), В.М. Матюшок, А. Одлизко (A. Odlyzko), С.И. Паринаова, А.Ю. Родионова, И.А. Стрелец, К. Шапиро (C. Shapiro), Н. Экономидиса (N. Economides) и др.

Вопросы потребительского качества и экономической эффективности информационных систем исследуются в трудах Б. Бозма (B. Boehm),

В.Н. Волковой, И.Н. Дрогобыцкого, Е.Н. Ефимова, В.В. Липаева, Л.Г. Матвеевой, К.Г. Скрипкина, Е.Н. Тищенко, Г.Н. Хубаева, В.Н. Юрьева и др.

Вопросы анализа, моделирования и совершенствования бизнес-процессов организации нашли отражение в работах А.М. Вендрова, О.В. Голосова, В.В. Дика, Е.Г. Ойхмана, Э.В. Попова, Е.В. Поповой, В.В. Репина, М. Робсона (M. Robson), Ю.Ф. Тельнова, М. Хаммера (M. Hammer), Дж. Харрингтона (J. Harrington), Дж. Чампи (J. Champy), А.-В. Шеера (A.-W. Sheer) и др.

Вопросы объектно-ориентированного моделирования и проектирования информационных систем анализируются в работах Г. Буча (G. Booch), Э. Гамма (E. Gamma), И. Грэхема (I. Graham), А.И. Долженко, Ф. Крачтена (Ph. Kruchten), Б. Мейера (B. Meyer), М. Пенкера (M. Penker), Дж. Рамбо (J. Rumbaugh), М. Фаулера (M. Fowler), С. Шлеера (S. Shleer), И.Ю. Шполянской, Х.-Э. Эрикссона (H.-E. Eriksson), А. Якобсона (I. Jacobson) и др.

Проблемы моделирования интернет-приложений разрабатываются в трудах М. Брамбилла (M. Brambilla), С. Кери (S. Ceri), Дж. Коналлена (J. Connalen), Н. Кох (N. Koch), А. Крауса (A. Kraus), Е. Мендес (E. Mendes), О. Пастора (O. Pastor), Г. Росси (G. Rossi), В. Пелечано (V. Pelechano), Д. Швайбе (D. Schwabe) и др.

Тематике имитационного моделирования экономических систем и бизнес-процессов посвящены работы Н.П. Бусленко, А.А. Емельянова, У. Кельтона (W. Kelton), В.Н. Томашевского, Дж. Форрестера (J. Forrester), Р. Шеннона (R. Shannon), Дж. Шрайбера (T. J. Schriber) и др.

Проблемы совместного использования языка UML и метода имитационного моделирования изучаются в работах Л. Арифа (L.B. Arief), С. Бальсамо (S. Balsamo), М. Марцолла (M. Marzolla), Д. Петриу (D.C. Petriu), Р. Пули (R. Pooley) и др.

В то же время, пока не разработаны теоретические и прикладные вопросы создания методологии и инструментария моделирования¹ процессов эксплуатации интернет-приложений, которые соответствовали бы следующим требованиям: позволяли учитывать затраты труда на создание и эксплуатацию интернет-приложения во всех вариантах его реализации; отражали стохастический характер процессов функционирования интернет-приложений; имели возможность интеграции с информационными моделями, используемыми при проектировании и разработке интернет-приложений; требовали минимальных затрат труда на освоение и использование.

Цель и задачи диссертационного исследования. Целью диссертационного исследования является развитие методологии и инструментария моделирования процессов эксплуатации интернет-приложений. Поставленная цель потребовала решения следующих задач диссертационного исследования:

- 1) Развитие теоретических и методологических основ экономикоматематического моделирования процессов использования интернет-приложений.
- 2) Разработка методов и инструментария выбора компонентов интернет-приложения.

¹ Модель - искусственный объект, представляющий собой отображение (образ) системы и ее компонентов. Считается, что М моделирует А, если М отвечает на вопросы относительно А. Здесь М - модель, А - моделируемый объект (оригинал). (Рекомендации по стандартизации. – М.: Госстандарт России, 2001.)

3) Разработка методики сравнительного анализа вариантов реализации интернет-приложения на основе процессно-статистического учета затрат трудовых ресурсов².

4) Создание концепции и метамодели интеграции визуального и имитационного моделирования интернет-приложений на основе языка UML.

5) Разработка метода и алгоритмического обеспечения для автоматизированного формирования имитационных моделей.

6) Создание инструментальной системы автоматизированного синтеза имитационных моделей процессов эксплуатации интернет-приложений.

Объект и предмет исследования. Объектом диссертационного исследования являются интернет-приложения, используемые предприятиями всех форм собственности. Предметом исследования выступают процессы использования интернет-приложений в деятельности предприятий и организаций.

Теоретическая база исследования. Теоретическую и методологическую базу исследования составляют научные труды российских и зарубежных ученых по экономико-математическому моделированию, статистическим методам, объектно-ориентированному моделированию и проектированию информационных систем. В проведенном исследовании использовались методы анализа и моделирования деловых процессов. Также применялись методы сравнения сложных систем по критерию функциональной полноты и методы имитационного моделирования сложных технических и социально-экономических систем.

В работе обобщены результаты исследований автора за 1998-2009 годы в области экономики сети Интернет, экономических аспектов создания и использования интернет-приложений, автоматизации процесса построения имитационных моделей.

Диссертационная работа выполнена в рамках пунктов Паспорта специальности 08.00.13 – математические и инструментальные методы экономики: 2.2 *«Конструирование имитационных моделей как основы экспериментальных машинных комплексов и разработка моделей экспериментальной экономики для анализа деятельности сложных социально-экономических систем и определения эффективных направлений развития социально-экономической и финансовой сфер»* и 2.6 *«Развитие теоретических основ методологии и инструментария проектирования, разработки и сопровождения информационных систем субъектов экономической деятельности: методы формализованного представления предметной области, программные средства, базы данных, корпоративные хранилища данных, базы знаний, коммуникационные технологии»*.

Эмпирическая база исследования. Эмпирической базой диссертационного исследования стали экспериментальные и статистические данные, полученные при построении и эксплуатации интернет-приложений различного масштаба и назначения в течение нескольких лет, а также статистические данные по исполнению бизнес-процессов в организациях, предприятиях и учреждениях различных направлений деятельности (высшее образование, государственное управление, производство, оптовая торговля и др.).

² Хубаев Г.Н. Калькуляция себестоимости продукции и услуг: процессно-статистический учет затрат // Управленческий учет. – 2009. – №2. – С. 35-46.

Научная новизна диссертационной работы. Научная новизна диссертационного исследования заключается в развитии методологии и инструментария экономико-математического моделирования процессов использования интернет-приложений. Научную новизну содержат следующие результаты:

1) Проведена адаптация методологии и инструментария анализа сложных систем по критерию функциональной полноты для решения задач сравнительной оценки и выбора интернет-приложений. При этом:

- предложена модификация метода сравнительного анализа сложных систем по критерию функциональной полноты. Особенности модификации состоят в формировании групп функций, учете зависимостей между функциями и совместимости анализируемых систем. Использование предложенного варианта модификации обеспечивает построение совокупности функций и программных компонентов интернет-приложения, отражающей требования заказчика к функциональной полноте.

- разработано инструментальное средство анализа сложных систем по критерию функциональной полноты, отличающееся реализацией предложенных теоретических положений и позволяющее автоматизировать процедуры сравнительного анализа и выбора интернет-приложений.

2) Предложена методика анализа вариантов реализации интернет-приложений на основе процессно-статистического учета затрат трудовых ресурсов при использовании интернет-приложений, включающая оценку частотных и временных характеристик отдельных операций и суммарных трудозатрат за период и позволяющая осуществлять сравнение вариантов реализации интернет-приложения.

3) Разработана концепция интеграции визуального и имитационного моделирования при оценке характеристик потребительского качества интернет-приложений. Ориентация на использование визуальной UML-модели интернет-приложения позволяет существенно снизить трудоемкость построения имитационной модели.

4) Построена метамодель, реализующая концепцию интеграции визуального и имитационного моделирования, включающая: количественные компоненты модели (переменные) для представления параметров моделируемой системы; диаграмму прецедентов для представления совокупности моделируемых процессов и спецификации их частотных характеристик; диаграмму деятельности для определения структуры процесса и спецификации его временных характеристик. Метамодель позволяет определить состав и взаимосвязи компонентов, используемых для представления процессов эксплуатации интернет-приложений при их моделировании.

5) Разработан метод автоматизированного синтеза имитационной модели, включающий: построение программного кода имитационной модели на основе компонентов метамодели; отображение компонентов метамодели в виде синтаксических конструкций программного кода; формирование структуры программного кода на основе отношений компонентов визуальной модели. Использование предложенного подхода обеспечивает сокращение затрат труда на реализацию моделирования.

6) Построено алгоритмическое обеспечение автоматизированного синтеза имитационной модели на основе расширенных UML-диаграмм. Предложенная совокупность алгоритмов, ориентированная на использование отношений агрегации и зависимости между компонентами метамодели для рекурсивного построения программного кода имитационной модели, позволяет реализовать метод автоматизированного синтеза имитационной модели.

7) Создано инструментальное средство автоматизированного синтеза имитационных моделей на основе языка UML (СИМ-UML), отличающееся реализацией предложенных теоретических положений и алгоритмов. Инструментальное средство позволяет: конструировать визуальную модель и определять количественные параметры; генерировать программный код имитационной модели; проводить имитационное моделирование, получать для подмножеств операций и в целом для процессов функционирования интернет-приложений статистические характеристики и гистограмму распределения трудовых и финансовых затрат на эксплуатацию интернет-приложения.

8) Построены визуальные и имитационные модели интернет-приложений электронной коммерции, выполненные на основе разработанных теоретических положений и позволяющие: оценивать затраты труда на эксплуатацию интернет-приложений; исследовать влияние различных параметров на величину затрат труда; получать затраты труда в разрезе операций, процессов и исполнителей; сравнивать различные варианты реализации интернет-приложения электронной коммерции.

Теоретическая и практическая значимость результатов исследования.

Теоретическая значимость диссертационного исследования состоит в развитии методологии моделирования интернет-приложений для поддержки принятия решений при их построении и использовании, в развитии методов имитационного моделирования деловых процессов.

Практическая значимость диссертационного исследования определяется тем, что его основные положения, выводы, рекомендации, модели, методы и алгоритмы создадут основу для принятия решений при использовании и развитии интернет-приложений в деятельности предприятий и организаций. Отдельные предложенные модели и методы могут использоваться при разработке экономических информационных систем разной направленности. Методология и инструментарий визуального и имитационного моделирования на основе языка UML является средством построения моделей деловых процессов в любых областях производства и управления для их анализа и совершенствования.

Апробация и внедрение результатов исследования. Основные результаты диссертационной работы докладывались и обсуждались на научно-практических конференциях и семинарах различного уровня, в том числе: «Инновации в науке и образовании-2008» (юбилейная международная научная конференция, г. Калининград, КГТУ, 2008); «Компьютерная техника и технологии» (региональная научно-техническая конференция, г. Ставрополь, СКГТУ, 2003); «Компьютерное моделирование 2008» (международная научно-практическая конференция, г. Санкт-Петербург, СПбГУ, 2008); «Математические и статистические методы в экономике и естествознании» (межвузовские научные чтения, г. Ростов-на-Дону, РГЭУ «РИНХ», 1999, 2003); «Моделирование. Теория, методы и средства» (VIII

международная научно-практическая конференция, г. Новочеркасск, ЮРГТУ (НПИ), 2008); «ИННОВ-2005» (выставка-ярмарка научно-технических разработок в рамках международного инновационного форума, г. Новочеркасск, ЮРГТУ (НПИ), 2005); «Новые технологии в управлении, бизнесе и праве» (III международная научно-практическая конференция, г. Невинномысск, НИУБ и П, 2003); «Проблемы информационной безопасности» (всероссийская научно-практическая интернет-конференция, г. Ростов-на-Дону, РГЭУ «РИНХ», 2006, 2007); «Проблемы создания и использования информационных систем и технологий» (межрегиональная научно-практическая конференция, г. Ростов-на-Дону, РГЭУ «РИНХ», 2007, 2008, 2009); «Проблемы теории и практики развития региональной статистики» (межрегиональная научная конференция, г. Ростов-на-Дону, РГЭУ «РИНХ», 2003); «Компьютерные технологии в науке, производстве, социальных и экономических процессах (КТ 2009)» (X международная научно-практическая конференция, г. Новочеркасск, ЮРГТУ (НПИ) 2009); «Системный анализ в проектировании и управлении» (X международная научно-практическая конференция, г. Санкт-Петербург, СПбГПУ, 2006); «Статистика в современном мире: методы, модели, инструменты» (Межвузовская научно-практическая конференция, г. Ростов-на-Дону, РГЭУ «РИНХ», 2007, 2008, 2009, 2010); «Теория, методы проектирования, программно-техническая платформа корпоративных информационных систем» (VI международная научно-практическая конференция, г. Новочеркасск, ЮРГТУ (НПИ) 2008); «Технологии информационного общества – Интернет и современное общество» (всероссийская объединенная конференция, г. Санкт-Петербург, СПбГУ, 2005, 2006, 2008); «Экономико-организационные проблемы проектирования и применения информационных систем» (международная научно-практическая конференция, г. Кисловодск, РГЭУ «РИНХ», 2005, 2007, 2008, 2010); «Экономико-организационные проблемы проектирования и применения информационных систем» (всероссийская научно-практическая конференция, г. Ростов-на-Дону, РГЭУ «РИНХ», 2000, 2001); «Экономические проблемы организации производственных систем и бизнес-процессов (ЭПО-2009)», (VII международная научно-практическая конференция, г. Новочеркасск, ЮРГТУ (НПИ), 2009).

Основные результаты диссертационного исследования используются в деятельности ряда организаций (ООО «Инфотэкс», ООО «ЕЙСК-ИНФО», ООО «Стэйт-Он Лаб» и др.) при создании и эксплуатации интернет-приложений, а также для совершенствования деловых процессов. Разработанные методы и инструментарий нашли применение в учебном процессе Ростовского государственного экономического университета (РИНХ), специальность «Прикладная информатика (по областям)» (дисциплины «Системный анализ», «Разработка системы web-представительства фирмы», «Имитационное моделирование экономических процессов»; дипломное проектирование), а также в ряде других вузов. Отдельные результаты диссертационной работы использованы для выполнения научно-исследовательских работ для Государственного научно-исследовательского института развития налоговой системы Федеральной налоговой службы России (х/д № 926/06, № 958/07-8-ЮР/С, № 959/07-9-ЮР/С).

Публикации. Основные результаты диссертационного исследования изложены в 55 научных работах, в том числе в 11 статьях в журналах из перечня изда-

ний, рекомендованных ВАК РФ для публикации основных результатов докторских диссертаций, двух монографиях и двух научных изданиях; получено 5 Свидетельств о государственной регистрации программ для ЭВМ. Общий объем авторских публикаций по теме диссертации 42,6 печатных листа.

Структура и объем работы. Диссертационная работа состоит из введения, пяти глав, заключения, библиографического списка и приложений. Библиографический список включает 302 литературных источника.

ОСНОВНЫЕ ПОЛОЖЕНИЯ ДИССЕРТАЦИИ

Во введении обосновывается актуальность рассматриваемой проблемы, формулируются цель и задачи диссертационного исследования, определяются его объект и предмет, рассматриваются теоретические и методологические основы исследования, выделяются научная новизна и практическая значимость работы, приводится оценка внедрения и апробации исследования, описывается структура диссертационной работы.

В первой главе «Экономические аспекты построения и использования интернет-приложений» выполнен анализ проблем исследования экономической стороны процессов использования интернет-приложений в деятельности предприятий и организаций.

Развитие сети Интернет и интернет-технологий потребовало осмысления и изучения связанных с ней социально-экономических процессов и явлений. В зарубежной и отечественной науке происходит обращение к вопросам исследования экономических аспектов влияния сети Интернет и интернет-технологий на деятельность организаций и на экономику в целом. В таблице 1 представлены основные направления исследования экономической стороны сети Интернет, разделенные по уровням.

Существующие экономические исследования преимущественно рассматривают хозяйственную деятельность, осуществляемую с помощью сети Интернет и современных информационных и коммуникационных технологий. При этом на первый план выходят вопросы оценки влияния сети Интернет на характер экономических отношений, а также ее воздействие на рынок, прежде всего, с точки зрения изменения транзакционных издержек. Имеется ряд работ, которые рассматривают экономические аспекты сети Интернет, на уровне сети передачи данных. В то же время, проблемы использования возможностей сети Интернет и интернет-приложений на уровне предприятия пока не нашли достаточного отражения в экономической науке.

Исследование экономических аспектов использования интернет-приложений в деятельности предприятия может опираться на современные достижения экономической науки, которые широко применяются для изучения сложных информационных систем. В частности, могут использоваться экономико-математические и статистические методы, метод имитационного моделирования, методы анализа экономической эффективности и потребительского качества информационных систем, методы анализа и моделирования деловых процессов.

Таблица 1

Основные направления исследования экономических аспектов сети Интернет

Уровень	Направления
Глобальный уровень	<ul style="list-style-type: none"> - влияние сети Интернет на мировую экономику; - информационная экономика (сеть Интернет как инфраструктура информационной экономики); - влияние сети Интернет на национальную экономику; - роль сети Интернет в процессах глобализации; - трансформация экономических отношений под влиянием компьютерных сетей и информационных технологий («новая экономика»).
Уровень сети Интернет	<ul style="list-style-type: none"> - свойства сети Интернет как экономической системы; - модели роста сети Интернет; - секторы экономики сети Интернет, балансовые модели; - графовые модели WWW, анализ структуры и динамики.
Уровень рынка	<ul style="list-style-type: none"> - влияние сети Интернет на рынок (уровень цен, дисперсия цен, конкуренция); - транзакционные издержки; - электронная коммерция.
Уровень предприятия	<ul style="list-style-type: none"> - оценка экономической эффективности интернет-приложений; - влияние сети Интернет на бизнес-процессы предприятия; - анализ рентабельности инвестиций в интернет-проекты; - оценка экономической эффективности интранет-сети; - выбор структуры, содержания и средств построения интернет-приложений.
Уровень сети передачи данных	<ul style="list-style-type: none"> - взаимодействие подсетей (interconnection); - стандартизация программных и технических средств; - алгоритмы организации передачи данных с применением экономических методов разделения ограниченных ресурсов; - модели ценообразования на телекоммуникационные услуги.

Интернет-приложения отличаются соотношением коммуникационной и информационной составляющей, характеризуются разными функциональными и нефункциональными требованиями. Оценка эффективности различных классов интернет-приложения основывается на разных критериях (затраты труда на создание, затраты труда на поддержку, скорость разработки, функциональная полнота, надежность, качество дизайна, частота обновления и т.д.). Разные интернет-приложения требуют разной структуры инвестиций (соотношение текущих и капитальных затрат, соотношение затрат на программное обеспечение и затрат на оплату труда).

Таким образом, для анализа процессов использования интернет-приложений и для разработки в этой области экономико-математических моделей и методов необходима классификация интернет-приложений. Целями классификации являются: систематизация знаний о технических и экономических аспектах интернет-приложений; обеспечение выбора методов построения и развития интернет-приложения, выбора модели жизненного цикла; оценка и сравнение текущего и желаемого состояния интернет-приложения.

Для достижения этих целей система классификации интернет-приложений должна удовлетворять ряду требований: ориентация на объективные классифика-

ционные признаки; покрытие всех стадий жизненного цикла интернет-приложений; отражение как технических, так и экономических особенностей интернет-приложений.

На рис. 1 представлена предложенная совокупность признаков классификации интернет-приложений.

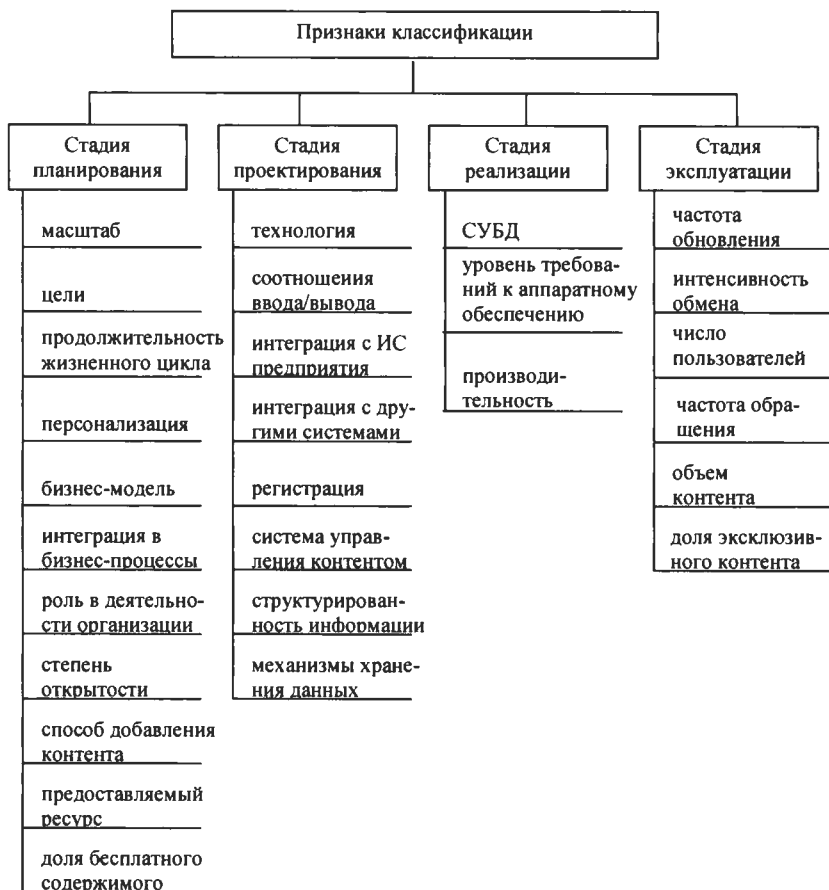


Рис. 1. Признаки классификации интернет-приложений

Большинство современных интернет-приложений отличают такие особенности, как: большое число компонентов; большое число пользователей; влияние разнообразных случайных факторов на функционирование интернет-приложения; значительное число технологий, используемых для построения приложения; сложный характер влияния на эффективность деятельности организации; включение интернет-приложения в различные бизнес-процессы организации; значительные затраты труда на создание и поддержку. Сложность интернет-приложений

создает необходимость обращения к методам моделирования при их построении, использовании и развитии.

Для формальной постановки задач построения интернет-приложения необходимо представить различные стороны интернет-приложений в терминах теории множеств. Интернет-приложение можно рассматривать как пятерку вида:

$$IS = \langle M, P, IT, R, F \rangle,$$

где M – модель интернет-приложения, представляющая в общем виде его цели и содержание;

P – проект приложения, описывающий структуру интернет-приложения;

IT – множество интернет-технологий, используемых для создания интернет-приложения;

R – компонент, характеризующий реализацию интернет-приложения;

F – компонент, характеризующий использование интернет-приложения.

Рассмотрим подробнее содержание отдельных элементов.

Модель («ядро») приложения

$$M = \langle Tg, U, BP, CM, STAT \rangle$$

отражает его назначение, возможности и содержание, а также место в деятельности организации. Модель может представлять существующее интернет-приложение, предлагаемый вариант его развития или новое интернет-приложение. Здесь:

Tg – совокупность целей интернет-приложения;

U – варианты использования интернет-приложения;

BP – совокупность бизнес-процессов, связанных с интернет-приложением;

CM – концептуальная модель интернет-приложения;

$STAT$ – совокупность количественных характеристик интернет-приложения.

Модель описывает наиболее стабильную часть приложения, его «ядро», которое может сохраняться при изменении технологий, дизайна и реализации.

Тройка:

$$U = \langle \{U_s\}, \{Fn\}, \{ \{U_s, Fn_h\}_h \} \rangle$$

описывает аудиторию интернет-приложения и его использование. $\{U_s\}, i = \overline{1, NUs}$ – это множество ролей пользователей, например: «Преподаватель кафедры», «Модератор», «Покупатель». Множество $\{Fn\}, j = \overline{1, NFn}$ – определяет в самом общем виде функции интернет-приложения. Соответственно множество $\{ \{U_s, Fn_h\}_h \}, k = \overline{1, NUF}, l = \overline{1, NUs}, h = \overline{1, NFn}$ описывает обращение пользователя некоторой категории к определенной функции интернет-приложения.

Описанный компонент соответствует диаграмме прецедентов языка UML, которая может эффективно использоваться для представления ролей пользователей (соответствуют акторам диаграммы) и вариантов использования интернет-приложения.

Следующий компонент модели определяет степень вовлечения интернет-приложения в бизнес-процессы организации:

$$BP = \langle \{bpr\}, \{op^*\}, \{ \{bpr, op'_h, Iv\}_h \} \rangle$$

Здесь $\{bpr\}, i = \overline{1, Nbp}$ – множество деловых процессов организации, на кото-

рые интернет-приложение может оказывать прямое или косвенное влияние. Операции этих бизнес-процессов описываются множеством $\{op_u^v\}, u = \overline{1, Nbp}, v = \overline{1, Nop''}$.

Тройка $(bpr_l, op_h^l, lv_j)_k$ показывает, каким образом интернет-приложение затрагивает операцию $op_h^l, h = \overline{1, Nop'}$ делового процесса $bpr_l, l = \overline{1, Nbp}$. Здесь $lv_j \in LV$ задает уровень использования интернет-приложения при выполнении операции. Множество уровней может быть описано, например, так $LV = \{lv_1, lv_2, lv_3\}$. Здесь:

lv_1 - «Интернет-приложение используется для поддержки выполнения операции» (например, исполнитель обращается к интернет-приложению для получения информации о надежности поставщика);

lv_2 - «Операция выполняется через интернет-приложение» (например, заказ на поставку товаров от филиала передается через web-форму);

lv_3 - «Бизнес-процесс полностью направляется и выполняется в рамках интернет-приложения».

Интернет-приложения становятся средством совершенствования бизнес-процессов организации, позволяя: сократить затраты труда на выполнение операций делового процесса; снизить число ошибок; ускорить исполнение делового процесса; допустить участие в деловом процессе представителей организационно-партерных и удаленных сотрудников; снизить значение географического фактора при организации делового процесса.

Концептуальная модель интернет-приложения:

$$CM = \langle E, A, RI, \{r_{ik}, e_i, e_j\}_k \rangle,$$

где $E = \{e_i\}, i = \overline{1, Ne}$ - множество сущностей предметной области, представленных интернет-приложением, например: «Студент», «Дисциплина», «Книга», «Товар». Множество $A = \{a_j\}, j = \overline{1, Ne}, j = \overline{1, Nea'}$ - описывает атрибуты сущностей; Множество $RI = \{r_{ik}\}, k = \overline{1, Nri}$ описывает отношения между сущностями, а множество элементов вида $(r_{ik}, e_i, e_j)_k$ связывает отношения и сущности.

Рассмотрим проект интернет-приложения. Выражение

$$P = \langle REQ, \{Com_j\}, \{Pg_k\}, \{Link_l\} \rangle$$

описывает приложение на этапе проектирования, оно соответствует проекту интернет-приложения до выбора интернет-технологий и до начала его реализации. Здесь $REQ = REQ' \cup REQ''$ - множество функциональных и нефункциональных требований к интернет-приложению. $\{Com_j\}, j = \overline{1, Ncom}$ - множество функциональных подсистем или сервисов интернет-приложения. Пара $\langle \{Pg_k\}, \{Link_l\} \rangle$ описывает навигационную структуру интернет-приложения, включая множество web-страниц $\{Pg_k\}, k = \overline{1, Npg}$ и множество ссылок между ними $\{Link_l | Link_l = (Pg', Pg', par_l), l = \overline{1, NLink}\}$.

Построенная формальная модель интернет-приложения дает возможность сформулировать последовательность задач, решаемых при построении интернет-приложения. Задача планирования интернет-приложения $f(M) \rightarrow \max, M \in U_M$. Здесь U_M - множество возможных вариантов построения интернет-приложения.

Вектор-функция $f(M) = (f^1(M), f^2(M), \dots)$ описывает совокупность критериев эффективности интернет-приложения.

Задача проектирования интернет-приложения $fe(P) \rightarrow \max, P \in U_p(M), f(P) \geq \bar{f}$. Необходимо построить такой проект P интернет-приложения, который предполагает реализацию уже имеющейся модели M . Вектор целевых функций трансформируется в вектор ограничений $\bar{f} = (\bar{f}_1, \bar{f}_2, \dots)$, которым должны удовлетворять параметры проекта $f(P) = (f_1(P), f_2(P), \dots)$. Каждый из проектов может обеспечить достижение всех целей модели, при этом проекты сравниваются по критерию экономической эффективности $fe(P) \rightarrow \max$.

Задача выбора технологий $TCO(IT) \rightarrow \min, IT \in MIT, f(IT) \geq f(P), req_i \in REQ(IT), \forall i = \overline{1, Nreq}$. Необходимо выбрать набор технологий минимальной стоимости, позволяющих реализовать заданный проект. В первом ограничении задается множество MIT допустимых наборов технических средств. Второе ограничение предполагает, что значения характеристик приложения (например, технической производительности), которые могут быть достигнуты с помощью набора интернет-технологий, не должны быть ниже требуемого набора характеристик проекта. Третье ограничение говорит о том, что требования проекта (прежде всего, функциональные) должны быть потенциально реализуемыми с помощью выбранного набора технологий. В качестве целевой функции используется TCO (Total Cost of Ownership, совокупная стоимость владения интернет-приложением). TCO включает такие компоненты, как: стоимость приобретения программных и аппаратных средств; стоимость обучения разработчиков; затраты труда на построение интернет-приложения; затраты труда на эксплуатацию интернет-приложения; затраты труда пользователей на работу с интернет-приложением.

Решение перечисленных задач требует привлечения методов моделирования. Использование комплекса экономико-математических и имитационных моделей позволяет принимать обоснованные с экономической точки зрения решения по созданию, проектированию и эксплуатации интернет-приложений.

Во второй главе «Модели и методы сравнительного анализа и выбора интернет-приложений по критерию функциональной полноты» описывается комплекс формализованных методов, моделей и инструментов позволяющих обеспечить выбор различных технологий, компонентов и функций интернет-приложения при его построении и использовании.

При создании интернет-приложения необходимо принять ряд решений, например: использование статических или динамических способов реализации; механизмы обновления содержимого; набор используемых интернет-технологий; формат и структура базы данных и т.д. Технологии и программные средства, используемые при разработке интернет-приложений обладают разными характеристиками в плане производительности, надежности, требований к оборудованию, сложности; имеют различную стоимость.

Большинство решений на стадии проектирования должно приниматься на основании требований к интернет-приложению и параметров его эксплуатации. Основными критериями являются: экономическая эффективность; функциональная полнота; надежность; безопасность; производительность; гибкость; своевре-

менность предоставления информации; корректность предоставляемой информации. Набор критериев и их относительная важность зависят от области использования разрабатываемого интернет-приложения.

Применение метода анализа сложных систем по критерию функциональной полноты дает возможность сравнительного анализа и выбора интернет-технологий для реализации интернет-приложения, программных инструментов разработки интернет-приложения, функций и программных компонентов интернет-приложения.

Сравнение интернет-приложений или их компонентов по критерию функциональной полноты³ предполагает последовательность шагов. Введем обозначения:

$S = \{S_j\}, (j = \overline{1, n})$ - множество сравниваемых систем;

$F = \{f_j\}, (j = \overline{1, m})$ - множество, составляющее словарь функций;

$P_{ik}^{11} = |S_i \cap S_k|$ - мощность пересечения систем S_i и S_k по функциям;

$P_{ik}^{01} = |S_k / S_i|$ и $P_{ik}^{10} = |S_i / S_k|$ - мощность разности соответствующих систем.

В качестве меры рассогласования между системами S_i и S_k используется величина $R_{ik} = P_{ik}^{01} / (P_{ik}^{11} + P_{ik}^{10})$. Для оценки степени поглощения системой S_k системы S_i служит величина $H_{ik} = P_{ik}^{11} / (P_{ik}^{11} + P_{ik}^{10})$. Оценка степени подобия систем проводится на основе меры подобия Жаккарда $G_{ik} = P_{ik}^{11} / (P_{ik}^{11} + P_{ik}^{10} + P_{ik}^{01})$. После выполнения соответствующих расчетов формируются матрицы $\|P_{ik}^{01}\|, \|R_{ik}\|, \|H_{ik}\|, \|G_{ik}\|$.

Рассчитанные матрицы преобразуются в логические матрицы P_0, R_0, H_0, G_0 , в соответствии с различными пороговыми значениями. Например, для матрицы подобия систем на основе порогового значения ε_g рассчитывается матрица $G^0 = \{G_{ij}^0\}$, где:

$$G_{ij}^0 = \begin{cases} 1, & \text{если } G_{ij} \geq \varepsilon_g, i \neq j \\ 0, & \text{если } G_{ij} < \varepsilon_g, \text{ или } i = j \end{cases}$$

Анализ матриц и графов позволяет исследовать совокупность изучаемых систем по критерию функциональной полноты.

Предложена модификация метода анализа сложных систем по критерию функциональной полноты ориентированная на отражение особенностей интернет-приложений. Расширения позволяют учесть: группы функций интернет-приложений; количественные и порядковые характеристики; технологии реализации приложений; зависимости между функциями; иерархию компонентов интернет-приложения.

³ Хубаев Г.Н. Сравнение сложных программных систем по критерию функциональной полноты // Программные продукты и системы (SOFTWARE & SYSTEMS). – 1998. – №2. – С. 6-9.

Разработана методика формирования структуры интернет-приложения на основе адаптированного метода анализа сложных систем по критерию функциональной полноты. Методика позволяет определить комплекс проектных решений, отражающих требования к функциональной полноте интернет-приложения. Рассмотрим последовательность шагов методики.

Шаг 1. Формируется перечень возможных функций планируемого интернет-приложения $F = \{f_j\}, j = \overline{1, m}$.

Источниками могут выступать результаты анализа существующих интернет-приложений, работающих в рассматриваемой или смежной с ней областях, интервью с заинтересованными лицами, результаты совещаний и т.д.

Шаг 2. Определяется множество проектов интернет-приложения $S = \{s_i\}, i = \overline{1, n}$. Проекты могут быть представлены разными организациями или подразделениями, могут соответствовать некоторым типовым решениям. Проекты отличаются по составу используемых сущностей предметной области, по организации процессов эксплуатации интернет-приложения. Соответственно, каждый проект реализует некоторое подмножество функций, что может быть отражено с помощью матрицы $X = \{x_{ij}\}, i = \overline{1, n}, j = \overline{1, m}$, где:

$$x_{ij} = \begin{cases} 1, & \text{если функция } j \text{ входит в систему } i \\ 0, & \text{если функция } j \text{ не входит в систему } i \end{cases}$$

Шаг 3. Производится сравнительный анализ проектов интернет-приложения в соответствии со стандартной методикой анализа программных систем по критерию функциональной полноты.

Далее в матрицу X добавляется дополнительная строка, соответствующая условной системе, которая отражает требования к функциональной полноте создаваемого интернет-приложения. Расчеты по матрице X повторяются. На рис. 2 приведен пример графа поглощения проектов интернет-приложения.

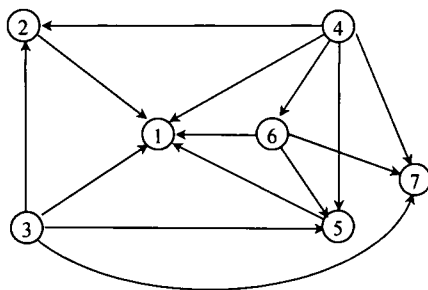


Рис. 2. Граф поглощения проектов интернет-приложения с включенной условной системой ($\epsilon_n=1$, полное поглощение)

В результате выполнения текущего шага методики необходимо выбрать те проекты, которые, с одной стороны, поглощают условную систему, с другой – близки к ней по функциональной полноте.

Шаг 4. Строится перечень программных систем, обеспечивающих построение интернет-приложения, и перечень функций этих систем (рис. 3). Для определенности будем говорить о системах управления содержимым (системах управления контентом, CMS), которые сегодня являются наиболее широко используемым средством построения интернет-приложений. Примерами распространенных систем могут служить 1С-Битрикс, Joomla!, Drupal и др.

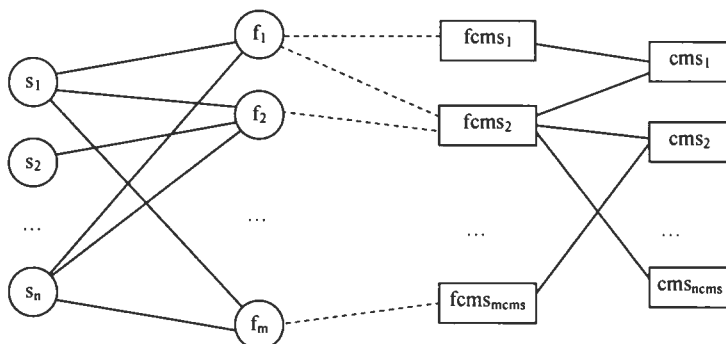


Рис. 3. Исходные данные для сравнительного анализа проектов интернет-приложений с учетом систем управления контентом

Формируются:

- множество систем управления контентом $CMS = \{cms_k\}, k = \overline{1, mcms}$;
- множество функций систем управления контентом $FCMS = \{fcms_h\}, h = \overline{1, mcms}$;

- матрица XCMS, элементы которой формируются следующим образом:

$$xcms_{ij} = \begin{cases} 1, & \text{если функция } j \text{ входит в } cms_i \\ 0, & \text{если функция } j \text{ не входит в } cms_i \end{cases}$$

В таблице 2 представлен фрагмент перечня функций систем управления контентом.

Таблица 2

Перечень функций систем управления контентом (фрагмент)	
Код	Функция
...	...
fcms11	Встроенный WYSIWG-редактор
fcms12	Встроенный редактор изображений
...	...
fcms45	Поддержка многоязычности
fcms46	Поддержка протокола SSL
fcms47	Поддержка фирмы производителя
fcms48	Подтверждение регистрации через E-mail
...	...

Проводится анализ систем управления контентом по критерию функциональной полноты. Например, на рис. 4 приведен пример графа подобия CMS.

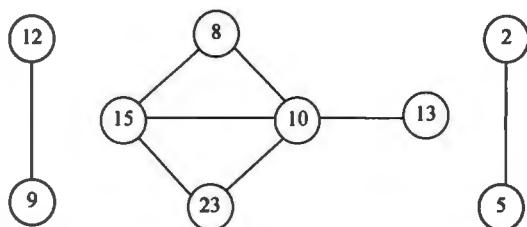


Рис. 4. Граф подобия систем управления контентом ($\varepsilon_g=0.80$)

Шаг 5. Для каждого из рассматриваемых проектов интернет-приложений формируется множество требований к системе управления контентом.

Таким образом, к множеству систем управления контентом добавляются условные системы $ucms_1 \dots ucms_n$, функции которых позволяют реализовать каждый из проектов интернет-приложения $s_1 \dots s_n$. Соответствующие строки добавляются в матрицу XCMS.

Шаг 6. Проводится анализ систем управления контентом с учетом условных систем.

По дополненной матрице XCMS строятся матрицы поглощения, подобия, превосходства и соответствующие графы. Таким образом, могут быть выделены системы, в необходимой степени поглощающие функции условной системы, и системы, которые в достаточной степени близки к условной.

В результате может быть сформирована матрица совместимости $C = \{c_{ik}\}$, $i = 1, n; k = 1, ncms$, элементы которой показывают, может ли проект интернет-приложения быть построен на базе некоторой CMS:

$$c_{ik} = \begin{cases} 1, & \text{если cms } k \text{ позволяет реализовать проект } i \\ 0, & \text{иначе} \end{cases}$$

Пример матрицы совместимости приведен в таблице 3.

Таблица 3

Пример матрицы совместимости проектов интернет-приложения и систем управления контентом (фрагмент)

	CMS1	CMS2	CMS3	CMS 4	CMS5	CMS6	CMS7	CMS8	CMS9	CMS10	
s3	1	1	0	0	0	0	0	1	1	1	...
s4	1	1	0	1	1	0	0	1	0	1	
s6	1	0	0	0	0	0	0	1	0	1	

Шаг 7. Производится объединение матрицы проектов и матрицы систем управления контентом.

При этом из декартова произведения множеств S и CMS исключаются варианты, не отражающие требования к совместимости:

$$S' = \{(s_i, cms_k) | c_{ik} = 1\}, i = \overline{1, n}; j = \overline{1, ncms}.$$

Множества функций объединяются: $F' = F \cup FCMS$.

В соответствии с полученными множествами S' и F' формируется матрица X' . Примерный вид матрицы X' приведен в таблице 4.

Таблица 4

Объединенная матрица вариантов реализации интернет-приложения

Вариант реализации		Функции проекта			Функции CMS		
Проект	CMS	f_1	...	f_m	f_{cms_1}	...	$f_{cms_{m_{cms}}}$
s_1	cms_1						
s_1	cms_5						
s_2	cms_3						
s_2	cms_5						
...							
s_0'							

Шаг 8. Формируется множество требований пользователя к системе управления контентом.

Эти требования описывают возможности создаваемого интернет-приложения с точки зрения удобства, гибкости, расширяемости и т.д. Например, «Использование свободно-распространяемой системы управления контентом».

Требования к системе управления контентом объединяются с требованиями к интернет-приложению. Сформированная строка матрицы X' описывает функциональный состав условной системы s_0' .

Шаг 9. Проводится сравнение вариантов реализации интернет-приложения.

Рассчитываются матрицы P , H , G и строятся соответствующие графы (например, на рис. 5 представлен граф подобия вариантов реализации интернет-приложения, а в таблице 5 – фрагмент матрицы поглощения).

Таблица 5

Пример матрицы поглощения для вариантов реализации интернет-приложения (фрагмент)

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	...
s1	1,000	0,845	0,905	0,845	0,929	0,810	0,988	0,833	0,798	0,869	0,893	
s2	0,922	1,000	0,922	0,896	0,948	0,844	0,909	0,987	0,805	0,948	0,909	
s3	0,916	0,855	1,000	0,831	0,988	0,771	0,904	0,843	0,771	0,855	0,988	
s4	0,947	0,920	0,920	1,000	0,933	0,813	0,933	0,907	0,827	0,893	0,907	
s5	0,876	0,820	0,921	0,787	1,000	0,753	0,865	0,809	0,742	0,831	0,910	
s6	0,971	0,929	0,914	0,871	0,957	1,000	0,957	0,914	0,857	0,914	0,900	
s7	0,965	0,814	0,872	0,814	0,895	0,779	1,000	0,849	0,814	0,884	0,907	
s8	0,886	0,962	0,886	0,861	0,911	0,810	0,924	1,000	0,823	0,962	0,924	
...												
s18	1,000	0,878	0,976	0,854	0,951	0,902	1,000	0,878	0,927	0,902	0,976	

Возможные варианты сравниваются по функциональной полноте и сопоставляются с условной системой, отражающей требования к создаваемому интернет-приложению.

На основе матриц и графов выбирается подмножество вариантов, соответствующее требованиям к функциональной полноте интернет-приложения.

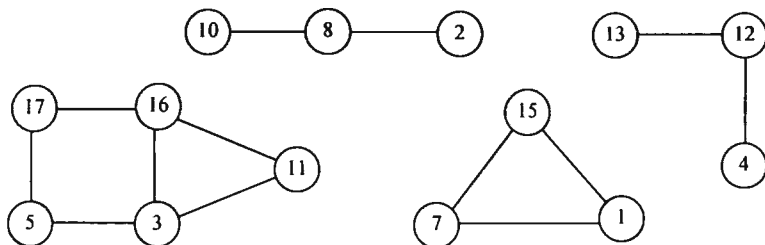


Рис. 5. Граф подобия вариантов реализации интернет-приложения ($\epsilon_B=0.90$)

В целом, описанная методика может быть наглядно представлена в виде схемы на рис.6.

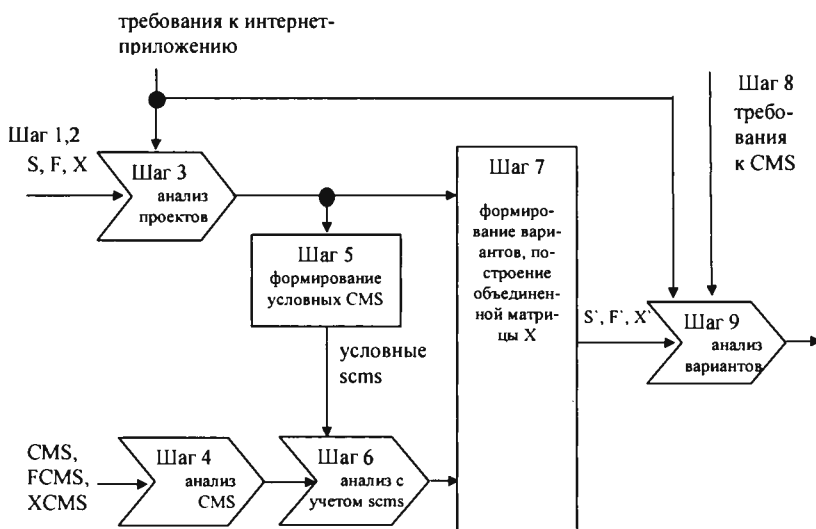


Рис. 6. Последовательность шагов формирования структуры интернет-приложения

Таким образом, рассмотренная адаптация метода анализа сложных систем по критерию функциональной полноты позволяет:

- сравнить различные проекты интернет-приложения с точки зрения функциональной полноты и сопоставить их с требованиями пользователя;
- построить перечень функций систем управления контентом для реализации интернет-приложения;
- провести сравнительный количественный анализ систем управления контентом по критерию функциональной полноты, выявить системы, превосходящие другие, построить группы подобных по функциональной полноте систем;
- сформировать перечень вариантов реализации интернет-приложения с

учетом использования систем управления контентом;

- провести сравнительный анализ вариантов реализации интернет-приложений с учетом требований к создаваемому интернет-приложению и требований к системам управления контентом.

Применение метода анализа сложных систем по критерию функциональной полноты к интернет-приложениям позволяет выбрать несколько возможных вариантов реализации интернет-приложения. Дальнейший анализ должен опираться на использование таких критериев, как стоимость, затраты труда на построение и эксплуатацию.

Разные варианты построения интернет-приложения характеризуются различной величиной расходов на его создание и поддержку. Решение о способе построения интернет-приложения принимается на основе критерия экономической эффективности в условиях конкретной задачи.

Для оценки экономической эффективности интернет-приложений можно использовать адаптированный подход профессора Г.Н. Хубаева⁴, основанный на оценке затрат труда на выполнение функциональных операций. С помощью экономико-статистического анализа оцениваются частотные и временные характеристики функциональных операций, выполняемых программной системой, и рассчитываются суммарные трудозатраты за период. Экономическая эффективность информационной системы определяется путем сопоставления экономии труда (в сравнении с базовым вариантом) и затрат на построение системы.

Сложность современных интернет-приложений и стохастический характер процессов их эксплуатации делает метод имитационного моделирования наиболее предпочтительным методом оценки затрат труда на построение и эксплуатацию. При этом набор визуальных моделей интернет-приложения может рассматриваться как основа для построения имитационной модели процессов эксплуатации интернет-приложения.

В третьей главе «Методология построения моделей процессов использования интернет-приложений» предложена концепция интеграции визуального и имитационного моделирования интернет-приложений на основе UML-моделей и описана разработанная совокупность модельных компонентов.

Управление информационной системой, подобной интернет-приложению, предполагает принятие решений в условиях действия большого числа внешних факторов, наличия множества взаимодействующих элементов управляемой системы и ориентировано на достижение комплекса различных целей. Средством поддержки принимаемых решений служит моделирование. Использование моделей позволяет рассмотреть различные аспекты объекта управления, исследовать последствия управляющих воздействий, сравнить различные варианты принимаемых решений.

Сегодня информационные системы все чаще рассматриваются в контексте деловых процессов, при этом проводится выделение деловых процессов, оценка эффективности их организации, исследование возможности их совершенствования. Анализ деловых процессов невозможен без использования того или иного

⁴ Хубаев Г.Н. Методика экономической оценки потребительского качества программных средств // Программные продукты и системы (SOFTWARE & SYSTEMS). – 1995. – №1. – С.2-8.

инструмента моделирования – общепризнанной нотации представления бизнес-процессов и методологии ее использования. Наиболее распространенными нотациями являются ARIS, семейство IDEF, BPMN, а также унифицированный язык моделирования UML.

Визуальное моделирование процессов эксплуатации интернет-приложений предоставляет возможности для их осмысления, оценки и рационализации.

Вместе с тем, переход на количественный уровень обеспечит более точную и обоснованную оценку существующего или возможного состояния системы, позволит выработать наиболее рациональный вариант совершенствования деловых процессов и построения информационной системы.

В большинстве случаев сложность изучаемой системы и наличие стохастической составляющей делают наиболее целесообразным использование имитационного моделирования. Имитационная модель обладает высокой степенью подобия моделируемой системе, позволяет рассматривать значительное число деталей и учитывает случайные факторы. Эксперименты над имитационной моделью способствуют получению оценки различных вариантов предлагаемых решений.

Недостатком имитационных моделей является трудоемкость их построения. При этом свойственная имитационным моделям привязка к конкретной системе и конкретным условиям влечет за собой необходимость многократной модификации модели.

За счет использования современных систем имитационного моделирования, ориентированных на создание модели в графическом режиме, можно сократить время и затраты труда на построение имитационной модели по сравнению с разработкой имитационных программ на языках высокого уровня или специализированных языках, подобных GPSS. Однако развитые современные системы имитационного моделирования дороги, достаточно сложны в освоении, требуют высокой и специализированной квалификации разработчика имитационных моделей. Для решения многих задач управления информационной системой зачастую не требуется привлечение всех функциональных возможностей системы имитационного моделирования и не всегда необходим предлагаемый системой уровень детализации.

Предлагаемая интеграция визуального и имитационного моделирования позволяет проводить одновременное исследование деловых процессов на качественном и количественном уровне, при этом визуальная модель служит основой для формирования структуры имитационной модели. Имитационное моделирование дает возможность рассматривать и сравнивать различные варианты построения системы, получать количественную оценку предлагаемых решений.

Проблема автоматизации построения имитационных моделей предполагает решение нескольких задач: выбор нотации и методики визуального моделирования; интеграция структурных и количественных компонентов; разработка методов и алгоритмов имитационного моделирования; создание соответствующего программного обеспечения.

Для решения задач имитационного моделирования интернет-приложений в качестве средства визуального представления деловых процессов был выбран

унифицированный язык моделирования UML, что обусловлено следующими преимуществами языка:

- 1) гибкостью и универсальностью. Средства языка можно использовать для решения задач анализа, моделирования и проектирования в различных областях;
- 2) возможностью отражения как статике, так и динамики моделируемой системы;
- 3) наличием представлений и визуальных средств, позволяющих рассматривать моделируемую систему: с разных сторон; на разном уровне детализации; на разных этапах анализа, проектирования и разработки;
- 4) ролью языка UML как общепризнанного стандарта проектирования и разработки программного обеспечения, включая интернет-приложения;
- 5) возможностью расширения, что позволяет адаптировать средства языка для эффективного решения задач моделирования в разных областях;
- 6) объектно-ориентированными принципами языка, соответствующими специфике имитационного моделирования.

Можно выделить следующие области пересечения тематики имитационного моделирования и применения языка UML, которые нашли отражение в работах отечественных и, прежде всего, зарубежных исследователей:

- представление системы имитационного моделирования;
- представление имитационной модели;
- использование существующих визуальных моделей на языке UML для поддержки или автоматизации построения имитационной модели.

Третье направление включает и вопросы применения UML-спецификаций программной системы для оценки ее технической производительности.

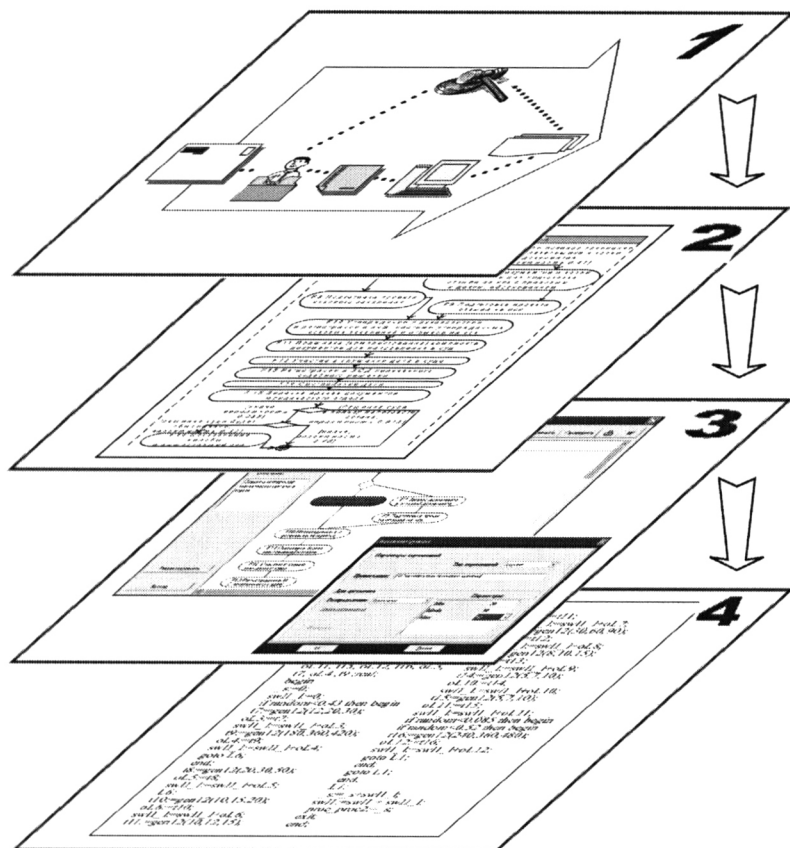
С другой стороны, UML-модели обладают значительным потенциалом в качестве средства представления деловых процессов при их имитационном моделировании, однако на сегодняшний день методологии имитационного моделирования деловых процессов на основе UML-моделей, ориентированной на автоматизированное формирование программного кода, не разработано.

Разработанная концепция интеграции визуального и имитационного моделирования предполагает, что:

- 1) с помощью выбранного подмножества диаграмм языка UML и их элементов описываются структурные и поведенческие аспекты моделируемой системы;
- 2) количественные компоненты (переменные имитационной модели) описывают частотные и временные параметры системы, с учетом их случайного характера;
- 3) диаграммы, элементы диаграмм и количественные компоненты связываются между собой в соответствии с заданными правилами;
- 4) такая взаимосвязанная совокупность визуальных и количественных компонентов служит основой для проведения имитационного моделирования.

Интеграция визуального и имитационного моделирования, во-первых, позволяет сократить затраты труда на построение имитационных моделей, во-вторых, дает возможность моделирования изучаемой системы на качественном и количественном уровне, в-третьих, позволяет снизить семантический разрыв между предметной областью и средствами ее представления в модели.

Плоскости, изображенные на рис. 7, соответствуют уровням представления делового процесса при моделировании: уровень предметной области; уровень формализованного описания деловых процессов; уровень представления средствами программной системы визуального и имитационного моделирования; уровень программного кода имитационной модели.



Источник: Широкова С.Н., Щербаков С.М. Возможности метода и программного комплекса автоматизированного синтеза имитационных моделей деловых процессов // Компьютерное моделирование 2008: Труды междунауч.-техн. конф., г. Санкт-Петербург, 24 - 25 июня 2008г./ СПб: Изд-во Политехнического университета, 2008. — С. 259-268.

Рис. 7. Уровни представления делового процесса

Выделенные в результате анализа предметной области деловые процессы (1-ый слой) описываются в нотации унифицированного языка моделирования UML (2-ой слой), обеспечивающего визуализацию структуры процессов. Для задания количественных параметров системы вводятся переменные различных типов (количественные компоненты модели).

Исходное концептуальное представление множества бизнес-процессов и спецификация количественных характеристик осуществляются с помощью диаграммы прецедентов. Для формализованного описания структуры каждого делового процесса применяется диаграмма деятельности.

Компоненты модели строятся и связываются между собой с помощью программной системы, включающей графический конструктор для построения UML-моделей (3-ий слой).

На основе совокупности компонентов модели осуществляется автоматический синтез программного кода имитационной модели (4-ый слой), причем для каждого компонента модели формируется соответствующий фрагмент программного кода.

Реализация концепции интеграции визуального и имитационного моделирования потребовала формирования конкретной совокупности визуальных и количественных компонентов, используемых для моделирования (метамодели интеграции визуального и имитационного моделирования).

На рис. 8 в формате семантической сети представлены основные модельные компоненты и связи между ними. Волнистая линия отделяет сущности предметной области от компонентов модели, при этом показано соответствие между элементами модели и изучаемой системы.

Разработана формализованная метамодель, которая позволяет однозначно определить состав, структуру и взаимосвязь визуальных и количественных компонентов модели.

Модель описывается конструкцией вида:

$$M = \langle V, D^{UC}, D^A \rangle.$$

Модель включает множество переменных и множество UML-диаграмм деятельности и прецедентов:

$V = V^A \cup V^F$ – множество переменных;

$V^A = \{v_i^a\}, i = \overline{1, I}$ – множество переменных-аргументов. I – общее количество переменных-аргументов в модели;

$V^F = \{v_j^f\}, j = \overline{1, J}$ – множество переменных-функций. J – общее количество переменных-функций в модели;

$D^{UC} = \{d_p^{uc}\}, p = \overline{1, P}$ – множество диаграмм прецедентов. P – общее количество диаграмм прецедентов в модели;

$D^A = \{d_k^a\}, k = \overline{1, K}$ – множество диаграмм деятельности. K – общее количество диаграмм деятельности в модели.

Переменная-аргумент v_i^a описывается парой вида $\langle ID_i, VALUE_i \rangle$, где ID_i – идентификатор переменной; $VALUE_i$ определяет способ получения значения переменной. $VALUE_i = \langle \xi_i, \overline{par_i} \rangle$, где $\overline{par_i}$ – вектор параметров, количество которых определяется законом распределения ξ_i .

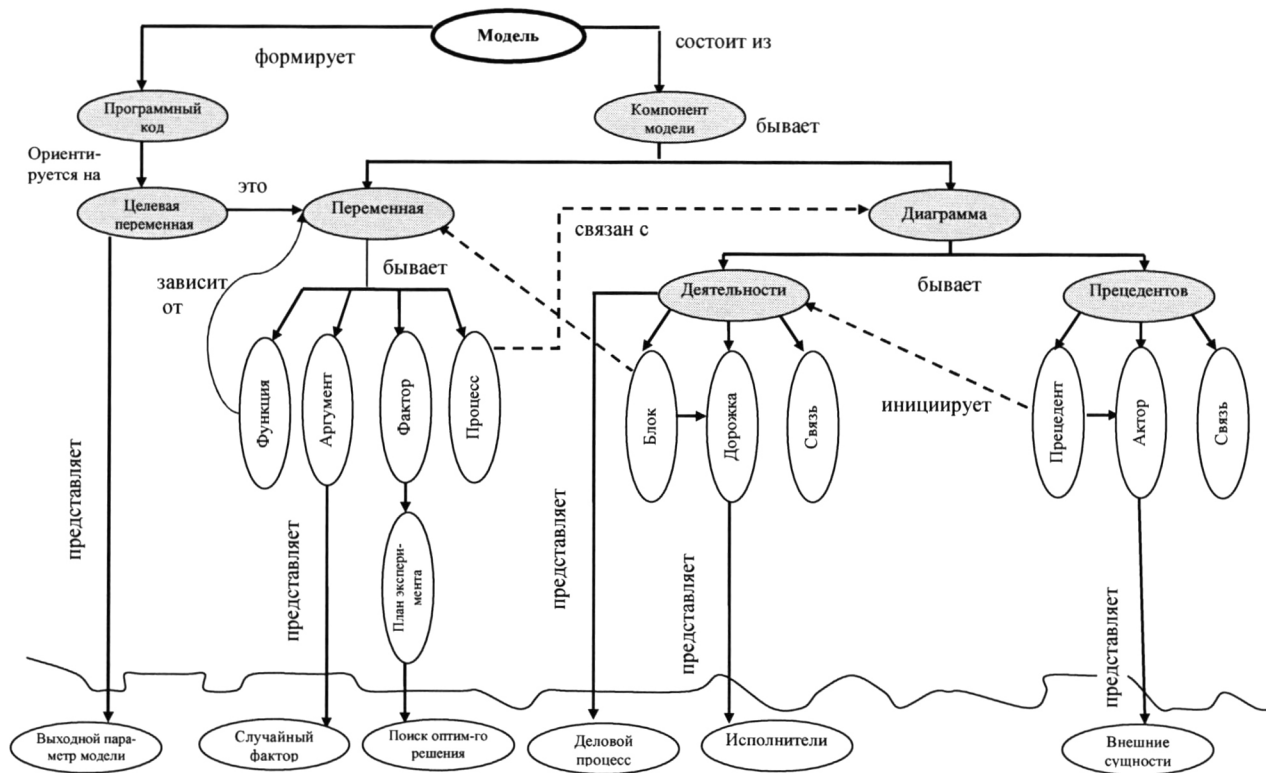


Рис.8. Компоненты метамодели интеграции визуального и имитационного моделирования интернет-приложений

При использовании табличного способа задания распределения: $VALUE_i = \{ \langle x_{iq}, f_{iq} \rangle, q = \overline{1, Q_i} \}$ – вектор пар вида $\langle x_{iq}, f_{iq} \rangle$, где x_{iq} – значение, f_{iq} – соответствующая вероятность, Q_i – число значений для i -ой переменной. Если используется непрерывный закон распределения: $VALUE_i = \{ \langle in_{iq}, f_{iq} \rangle, q = \overline{1, Q_i} \}$ – вектор пар вида $\langle in_{iq}, f_{iq} \rangle$, где in_{iq} – интервал, f_{iq} – соответствующая вероятность, Q_i – число интервалов для i -ой переменной.

Переменная-функция v_j^f описывается тройкой вида $\langle ID_j, F_j(V), G_j(V) \rangle$, где ID_j – идентификатор переменной; $F_j(V)$ – выражение, задающее взаимосвязь переменной v_j^f с другими переменными; $G_j(V)$ – выражение, описывающее верхний предел суммирования.

Диаграмма прецедентов d_p^{uc} задается парой $\langle AC_p, UC_p \rangle$, где:

$AC_p = \{ ac_{pw} \}, w = \overline{1, W_p^{AC}}$ – множество акторов диаграммы d_p^{uc} . W_p^{AC} – общее количество акторов p -й диаграммы прецедентов d_p^{uc} .

$UC_p = \{ uc_{pw} \}, w = \overline{1, W_p^{UC}}$ – множество прецедентов диаграммы d_p^{uc} . W_p^{UC} – общее количество прецедентов p -й диаграммы прецедентов d_p^{uc} .

Актор может быть задан конструкцией вида:

$$ac_{pw} = \langle CH_{pw}, \{ al_{pwh} \}, gn_{pw} \rangle,$$

где CH_{pw} – численность экземпляров акторов данного вида;

$\{ al_{pwh} \}, h = \overline{1, H_{pw}^{al}}$ – множество исходящих ассоциативных связей актора ac_{pw} ;

gn_{pw} исходящая связь типа «генерализация» актора ac_{pw} .

Ассоциативная связь задается парой $al_{ipw} = \langle v_{ipw}, auc_{ipw} \rangle$, где v_{ipw} – частота обращения к прецеденту по данной ассоциативной связи, а $auc_{ipw} \in UC_p$ – прецедент, на который указывает связь.

Связь типа «генерализация» $gn_{pw} = \langle ac_{ipw} \rangle$, где ac_{ipw} – актор-родитель для актора ac_{pw} . Связь может отсутствовать.

Прецедент может быть задан конструкцией вида:

$$uc_{pw} = \langle d_{pw}^A, \{ el_{pwh} \}, \{ il_{pwh} \} \rangle,$$

где $d_{pw}^A \in D^A$ – диаграмма деятельности, которая связана с данным прецедентом. Диаграмма деятельности описывает деловой процесс, который инициируется прецедентом;

$\{ el_{pwh} \}, h = \overline{1, H_{pw}^{el}}$ – множество исходящих связей типа «extend» прецедента uc_{pw} , H_{pw}^{el} – общее число исходящих связей прецедента uc_{pw} .

Каждая связь описывается парой: $el_{pwh} = \langle \varphi_{pwh}, luc_{pwh} \rangle$, где: φ_{pwh} – вероятность обращения по данной связи типа «extend»; $luc_{pwh} \in UC_p$ – прецедент, на который указывает связь.

$\{il_{pw,h}\}, h = \overline{1, H_{pw}^{il}}$ – множество исходящих связей типа «include» прецедента $ис_{pw}$, H_{pw}^{il} – общее число исходящих связей прецедента $ис_{pw}$. Элемент $il_{pw,h} \in UC_p$ показывает прецедент, с которым данный прецедент $ис_{pw}$ связан отношением «include».

Диаграмма деятельности задается конструкцией вида:

$$d_k^A = \langle bs_k, bf_k, BO_k, BC_k, BSA_k, SWL_k, PAR_k \rangle,$$

где: bs_k – блок начала диаграммы деятельности d_k^A ;

bf_k – блок окончания диаграммы деятельности d_k^A ;

$BO_k = \{bo_{ki}\}, t = \overline{1, TBO_k}$ – множество блоков операций диаграммы деятельности d_k^A ;

$BC_k = BC_k^C \cup BC_k^P$ – множество блоков ветвления диаграммы деятельности d_k^A ; $BC_k^C = \{bc_{ki}^C\}, t = \overline{1, TBCC_k}$ – множество блоков условия диаграммы d_k^A ;

$BC_k^P = \{bc_{ki}^P\}, t = \overline{1, TBPCP_k}$ – множество блоков вероятностных переходов диаграммы деятельности d_k^A ;

$BSA_k = \{bsa_{ki}\}, t = \overline{1, TBSA_k}$ – множество блоков подпроцесса диаграммы d_k^A ;

$SWL_k = \{swl_{ki}\}, n = \overline{1, NSWL_k}$ – множество дорожек диаграммы d_k^A .

$PAR_k = \{par_{kq}\}, q = \overline{1, NPAR_k}$, – множество переменных-параметров диаграммы деятельности d_k^A .

Обозначим через $B_k = \{bs_k\} \cup BO_k \cup BC_k \cup BSA_k \cup \{bf_k\}$ – множество блоков диаграммы деятельности d_k^A .

Блок операции bo_{ki} задается парой вида $bo_{ki} = \langle v_{ki}, nl_{ki} \rangle$, где: $v_{ki} \in V$ – переменная, определяющая продолжительность выполнения операции; $nl_{ki} \in B_k$ – связь, указывающая на следующий блок диаграммы деятельности d_k^A .

Блок условия bc_{ki}^C задается тройкой вида $bc_{ki}^C = \langle FC_{ki}(V), nyl_{ki}, nnl_{ki} \rangle$, где $FC_{ki}(V)$ – выражение, определяющее условие перехода по одной из двух исходящих связей. Исходящие связи определяют переходы на следующие блоки $nyl_{ki}, nnl_{ki} \in B_k$, соответственно для случаев, когда условие выполняется или не выполняется.

Блок вероятностных переходов bc_{ki}^P задается тройкой вида $bc_{ki}^P = \langle PC_{ki}(V), nxl_{ki}, nzl_{ki} \rangle$, где $PC_{ki}(V)$ – вероятность перехода на следующий блок по исходящей связи nxl_{ki} . Исходящие связи определяют переходы на следующие блоки $nxl_{ki}, nzl_{ki} \in B_k$.

Блок подпроцесса bsa_{ki} задается парой вида $bsa_{ki} = \langle d_{ki}^A, nl_{ki} \rangle$, где $d_{ki}^A \in D^A$ – диаграмма деятельности, описывающая дочерний процесс, связанный с данным блоком подпроцесса, связь $nl_{ki} \in B_k$ – обозначает следующий блок диа-

граммы деятельности d_k^A .

Дорожка swl_{kn} определяется парой $swl_{kn} = \langle IDSWL_{kn}, \{b_{knl}^{swl}\} \rangle$, где: $IDSWL_{kn}$ – идентификатор дорожки в модели; $\{b_{knl}^{swl}\}_{l=1, \overline{L_{kn}^{swl}}}$ – множество блоков $b_{knl} \in B_k$, относящихся к дорожке swl_{kn} .

Перечисленные компоненты модели могут использоваться для представления деловых процессов и служат основой для автоматизированного синтеза имитационной модели.

Рассмотрим роль отдельных компонентов модели для представления различных сторон моделируемых деловых процессов. Диаграммы языка UML описывают структуру и динамику изучаемых деловых процессов. Переменные представляют некоторые параметры изучаемой системы, например, частотные или временные. Переменная модели идентифицируется с помощью уникального, в рамках модели, имени переменной.

В модели используются различные виды переменных. Переменные-аргументы представляют в модели действие случайных факторов, оказывающих влияние на изучаемую систему, таких, как: время выполнения единичной операции, число позиций в поступившем заказе на производство и т.д. Переменная-функция зависит от других переменных, в том числе других переменных-функций. Примеры переменных-функций: прибыль, затраты труда на исполнение делового процесса за период времени, сумма налога.

Диаграмма деятельности (Activity Diagram) языка UML описывает деловой процесс. Диаграмма задает операции делового процесса, их последовательность и исполнителей, определяет возможные варианты исполнения процесса. Включенные количественные компоненты диаграммы деятельности позволяют описать количественную сторону исполнения делового процесса. Пример диаграммы делового процесса приведен на рис. 9. Здесь представлены блоки различных видов и дорожки. В виде комментариев языка UML показаны значения времени исполнения операций.

Блок операции (Activity) описывает операцию делового процесса, позволяет моделировать какую-либо деятельность, занимающую определенное время: технологическую операцию, обработку документа и т.п. С блоком операции связывается переменная, значение которой соответствует затратам труда на выполнение этой операции или ее стоимости. Так может быть учтен случайный характер времени выполнения операции.

Блок условия (Decision) позволяет моделировать альтернативные варианты исполнения делового процесса с помощью ветвлений и циклов.

Блок подпроцесса (Subactivity) позволяет организовать иерархию процессов. Например, при заключении договора осуществляется проверка контрагента. При каждом обращении к блоку подпроцесса будет запускаться на исполнение вложенный процесс. Использование вложенных деловых процессов повышает гибкость создаваемых моделей. Для блока подпроцесса задается имя переменной-процесса. Время исполнения этого процесса учитывается при определении времени исполнения родительского процесса.

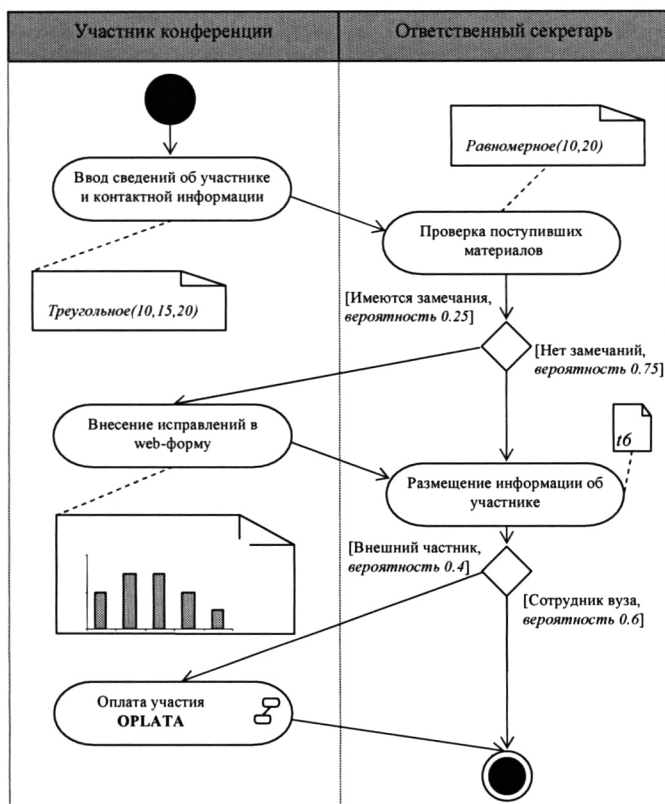


Рис. 9. Пример процесса функционирования интернет-приложения (в виде диаграммы деятельности)

Плавательные дорожки (Swimlanes) позволяют отслеживать затраты труда и стоимость не только по деловому процессу в целом, но и по его исполнителям. Например, если деловой процесс затрагивает несколько отделов, дорожки позволяют найти затраты труда/стоимость по каждому из этих отделов за заданный период времени.

Диаграмма прецедентов (Use Case Diagram) позволяет моделировать исследуемую совокупность бизнес-процессов в целом. Компоненты диаграммы прецедентов описывают границы системы, объединяют исследуемые деловые процессы, обеспечивают возможность моделирования на визуальном и на количественном уровне входной нагрузки изучаемой системы. Пример диаграммы прецедентов приведен на рис. 10.

Актор (Actor) представляет внешнюю по отношению к изучаемой системе сущность, порождающую разнообразные обращения к системе и вызывающую ее реакцию. Примерами акторов могут служить: заказчик, руководитель предпри-

ятия, и т.д. Для интернет-приложений акторами, как правило, выступают пользователи. С актором связывается переменная, определяющая численность его экземпляров.

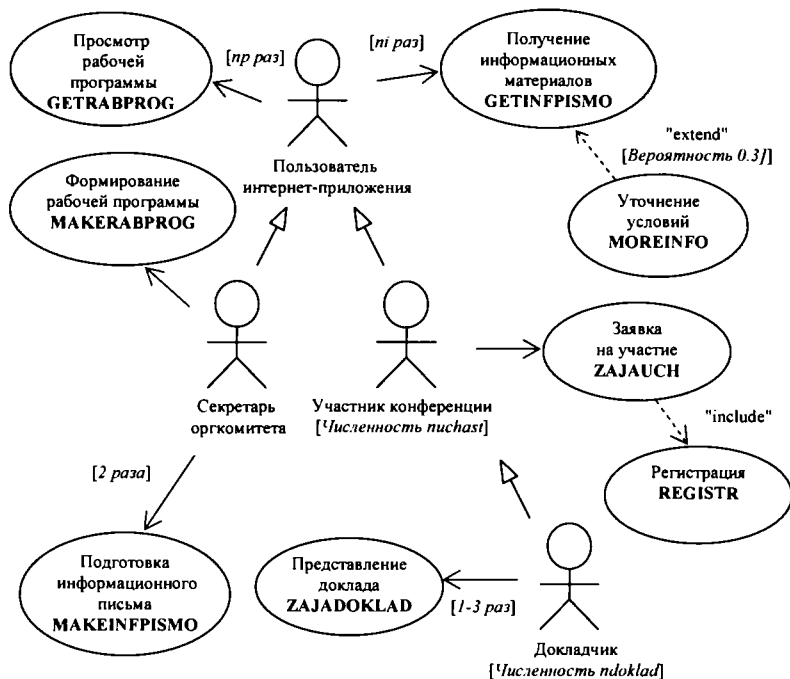


Рис. 10. Пример совокупности процессов функционирования интернет-приложения, описанных диаграммой прецедентов (фрагмент)

Прецедент (Use Case) описывает способ взаимодействия актора с системой. Примеры прецедентов: оплатить участие в конференции, заказать партию товара и т.д. Согласно метамодели, прецедент связан с некоторым деловым процессом и при обращении к прецеденту начинается исполнение этого делового процесса.

Ассоциация (Association) связывает актора с прецедентом. Ассоциации поставлена в соответствие переменная, описывающая частоту обращения актора к прецеденту за заданный период времени. Используются связи вида «include» и «extend» (для последней задается вероятность обращения к расширяющему прецеденту). Обобщение (Generalization) позволяет описать наследование акторов. Для актора-наследника будет производиться обращение ко всем прецедентам, связанным с актором-родителем.

В таблице 6 представлен пример перечня переменных имитационной модели, демонстрирующий взаимосвязь количественных и структурных компонентов.

Таблица 6

**Количественные компоненты имитационной модели функционирования
интернет-приложения (фрагмент)**

Имя	Вид переменной	Название	Тип / Закон распределения	Параметры / Формула расчета
nuchast	Аргумент	Число уч-ков без докладов	Нормальное	$\mu = 50, \sigma = 10$
nzaja	Аргумент	Число докладов	Нормальное	$\mu = 60, \sigma = 10$
...				
conf	Диаграмма прецедентов	Поддержка проведения конференции		
	Актор Аргумент	Секретарь оргкомитета	Число	значение = 1
	Актор Функция	Докладчик	Формула	nzaja
	...			
	Прецедент	Просмотр состояния	Процесс	analizsost
	Прецедент	Получение информационных материалов	Процесс	getinfpismo
	Прецедент	Заявка на участие	Процесс	zajauch
	Прецедент	Представление доклада	Процесс	zajadoklad
	Прецедент	Формирование рабочей программы	Процесс	makerabprog
	Прецедент	Регистрация	Процесс	registr
	...			
	Ассоциация Аргумент	Участник конференции / Заявка на участие	Число	значение = 1
	Ассоциация Аргумент	Руководитель оргкомитета / Просмотр состояния	Нормальное (усеченное)	$\mu = 30, \sigma = 10$
	Ассоциация Аргумент	Секретарь оргкомитета / Формирование рабочей программы	Число	значение = 1
	...			
zajadoklad	Диаграмма деятельности	Представление доклада на конференцию		
	Аргумент	Ввод сведений об авторе/соавторе	Треугольное	min = 7 moda = 10 max = 12
	Аргумент	Регистрация доклада и его загрузка	Треугольное	min = 10 moda = 15 max = 20
	Блок подпроцесса	Оплата участия	Дочерний процесс	oplata
	...			
oplata	Диаграмма деятельности	Оплата участия в конференции		
	Аргумент	Размещение реквизитов оплаты	Равномерное (min/max)	min = 3 max = 7
	...			
...				

В числе переменных показаны диаграммы языка UML и переменные, связанные с элементами диаграмм. Подобно переменной, диаграмма имеет имя для идентификации и при моделировании приобретает некоторое значение. Например, для диаграммы деятельности это значение может соответствовать затратам труда на исполнение делового процесса.

Входящий в метамодель набор компонентов может быть расширен, например, имеются возможности: подключения новых диаграмм языка UML и их элементов; создания новых типов связей между компонентами; введения новых видов количественных компонентов. Таким образом, описанный подход обладает открытостью, гибкостью и имеет потенциал для дальнейшего развития.

Инструментарий интеграции визуального и имитационного моделирования на основе языка UML может использоваться для оценки затрат труда на эксплуатацию интернет-приложений.

В четвертой главе «Разработка инструментария оценки затрат труда на эксплуатацию интернет-приложений» рассмотрены проблемы проведения имитационного моделирования на основе совокупности визуальных и количественных модельных компонентов, описаны метод автоматизированного синтеза имитационных моделей на основе UML-диаграмм и соответствующее алгоритмическое обеспечение. Также рассмотрены вопросы программной реализации концепции интеграции визуального и имитационного моделирования.

Использование выделенной совокупности визуальных и количественных компонентов для имитационного моделирования может быть обеспечено двумя способами: непосредственное имитационное моделирование или автоматизированный синтез имитационной модели. Второй подход отличается большей гибкостью и простотой реализации.

Разработанный метод автоматизированного синтеза имитационных моделей предполагает реализацию следующих принципов:

- 1) формирование программного кода имитационной модели на основе взаимосвязанных диаграмм языка UML и количественных компонентов (переменных);

- 2) соответствие между компонентами модели и фрагментами программного кода. Каждый такой фрагмент зависит от вида компонента, от его атрибутов и связей с другими компонентами. Содержимое фрагмента программного кода определяется специфичным для него алгоритмом имитационного моделирования;

- 3) отображение компонентов имитационной модели в виде синтаксических конструкций программного кода. Так, переменным модели соответствуют переменные в формируемой программе, а диаграммам языка UML – функции (подпрограммы). Зависимости между компонентами отражаются в виде вызовов функций и в виде операторов в синтаксисе выбранного языка программирования;

- 4) формирование структуры программного кода на основе структуры компонентов модели. Структура диаграмм UML и взаимосвязи компонентов служат основой структуры программного кода имитационной модели. Например, диаграмма деятельности определяет последовательность и состав команд в соответствующей функции, представляющей деловой процесс (прямой перенос логики процесса);

5) рекурсивное построение программного кода имитационной модели, начиная с целевой переменной (представляющей выходной параметр модели).

Преимущество реализации перечисленных принципов заключается в использовании возможностей языка UML по представлению динамики деловых процессов – UML-диаграммы определяют последовательность программных компонентов.

Еще одним преимуществом является гибкость – возможность изменять содержимое каждого фрагмента кода без изменения основных принципов метода и алгоритмов его реализации. Таким образом, имеется потенциал расширения функциональных возможностей имитационной модели.

На рис. 11 представлены некоторые компоненты модели и соответствующие участки программного кода имитационной модели (для иллюстрации использован синтаксис языка программирования Pascal).

Генерация программного кода по переменной зависит от вида переменной и от ее атрибутов. В сгенерированном программном коде имитационной модели переменная системы представляется как переменная языка Pascal (описывается в разделе `var` и используется в программе). Для расчета значения переменной формируется соответствующий программный код.

При построении программного кода производится определение переменных, участвующих в формировании значения целевой переменной, затем осуществляется рекурсивный вызов алгоритма генерации программного кода для каждой из этих переменных. Особым видом переменной является переменная-процесс. Каждой переменной-процессу поставлена в соответствие диаграмма языка UML. На основе диаграммы вычисляется некоторое значение, которое может трактоваться, например, как затраты труда на выполнение делового процесса. Это значение и становится значением переменной-процесса. В программном коде при обращении к переменной-процессу следует вызов функции, сгенерированной на основе UML-диаграммы.

При построении программного кода имитационной модели по диаграмме деятельности используется механизм прямого переноса логики диаграммы в программный код модели. Таким образом, алгоритм моделирования определяется структурой диаграммы. Диаграмме прецедентов также соответствует функция, которая при этом включает вложенные функции для каждого из акторов или прецедентов диаграммы.

Реализация метода автоматизированного синтеза потребовала разработки совокупности алгоритмов для формирования программного кода имитационной модели и для проведения имитационного моделирования. Полностью алгоритмическое обеспечение может быть представлено в виде схемы на рис. 12.

Разработана совокупность алгоритмов, в соответствии с которыми исполняется сама имитационная программа. Эти алгоритмы, реализованные для каждого из компонентов (диаграмма прецедентов, диаграмма деятельности, актор, прецедент, блок операции диаграммы деятельности, переменные различных видов), обеспечивают имитационное моделирование по этому компоненту и получение выходного значения. Предложены алгоритмы автоматического формирования программного кода имитационной модели для всех компонентов на основе диаграмм языка UML.

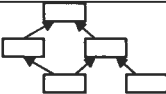
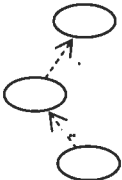
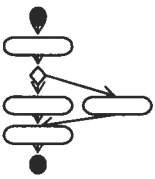
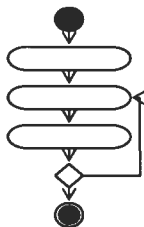
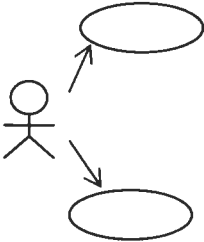
Модель	Код	Модель	Код
Переменные модели: x,y,z	<pre>var x:real; y:real; z:real;</pre>		<pre>X:=gen3(1,5); Y:=gen2(10,3); Z:=X*Y; H:=gen5(0.3); L:=H+Z;</pre>
	<pre>function _uc4:real; begin ... end; function _uc1:real; begin _uc1:=bp1; if random < 0.3 then _uc1:=_uc1+_uc4; _uc1:=_uc1+_uc3; end;</pre>	$S = \sum_{i=1}^n T_i$	<pre>N:=gen3(10,20); S:=0; for _i1:=1 to round(N) do begin T:=gen5(1.5); S:=S+T; end;</pre>
	<pre>function bp2:real; label L4; var _s:real; begin _s:=0; _s:=_s+gen1(1,3); if random<0.05 then begin _s:=_s+gen6(1,0.3); goto L4; end; _s:=_s+gen1(2,5); L4: _s:=_s+gen1(1,2); bp2:=_s; end;</pre>		<pre>{uml} function bp3:real; var _s:real; _il5:integer; n:real; begin _s:=0; _s:=_s+gen1(2,4); n:=gen12(3,5,10); for _iL5:=1 to round(n) do begin _s:=_s+gen1(1,3); _s:=_s+gen1(2,5); end; bp3:=_s; end;</pre>
	<pre>function ucd_ucc:real; var _s:real;_nI:real; _iI:integer; function _actor0:real; var _nII:real; _iII:integer; _sII:real; begin _sII:=0; _nII:=gen1(5,10); for _iII:=1 to round(_nII) do _sII:=_sII+_uc1; _nII:=gen1(10,15); for _iII:=1 to round(_nII) do _sII:=_sII+_uc2; _actor0:=_sII; end; begin _s:=0; _nI:=gen1(10,20); for _iI:=1 to round(_nI) do _s:=_s+_actor0; ucd_ucc:=_s; end;</pre>		

Рис. 11. Формирование программного кода по компонентам модели

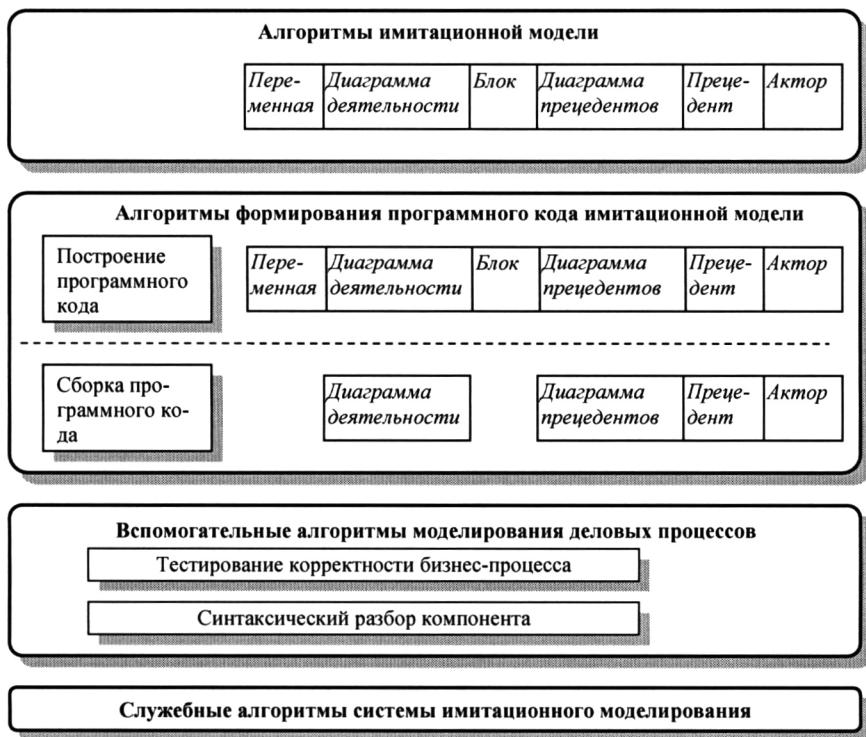


Рис. 12. Алгоритмы синтеза имитационных моделей

Среди вспомогательных алгоритмов моделирования деловых процессов можно выделить алгоритмы проверки корректности делового процесса. Проверяется допустимость процесса с содержательной точки зрения (например, корректность разделений/слияний, корректность циклов), а также корректность интеграции визуальных и количественных компонентов. Служебные алгоритмы обеспечивают функционирование системы имитационного моделирования, реализуя, например, создание диаграмм в графическом конструкторе.

Описанная структура алгоритмического обеспечения отличается достаточной гибкостью, оставаясь открытой для расширения. Существует возможность добавления новых компонентов и функций, при этом механизмы формирования программного кода (второй уровень) будут затронуты только частично.

Центральное место в алгоритмическом обеспечении занимает совокупность алгоритмов формирования программного кода имитационной модели. Для каждого компонента по определенным правилам строится фрагмент программного кода. В ряде случаев необходима сборка – повторный проход по компонентам и присоединение сформированных для них фрагментов программного кода к ос-

новой программе. Таким образом, создается программный код имитационной модели.

Между компонентами имитационной модели могут существовать отношения двух типов:

- отношения агрегации (компонент, включающий в себя другие компоненты, будем называть «композиционным», а входящие в его состав компоненты – «агрегированными»);

- отношения зависимости (значение компонента может рассчитываться на основании других компонентов, которые будем называть «влияющими»).

В качестве примеров отношений агрегации можно назвать следующие: диаграмма деятельности включает блоки; блок операции включает переменную; диаграмма прецедентов включает акторов, прецеденты и связи.

Для формирования программного кода по композиционному компоненту необходимо провести формирование программного кода по всем входящим в него агрегированным компонентам.

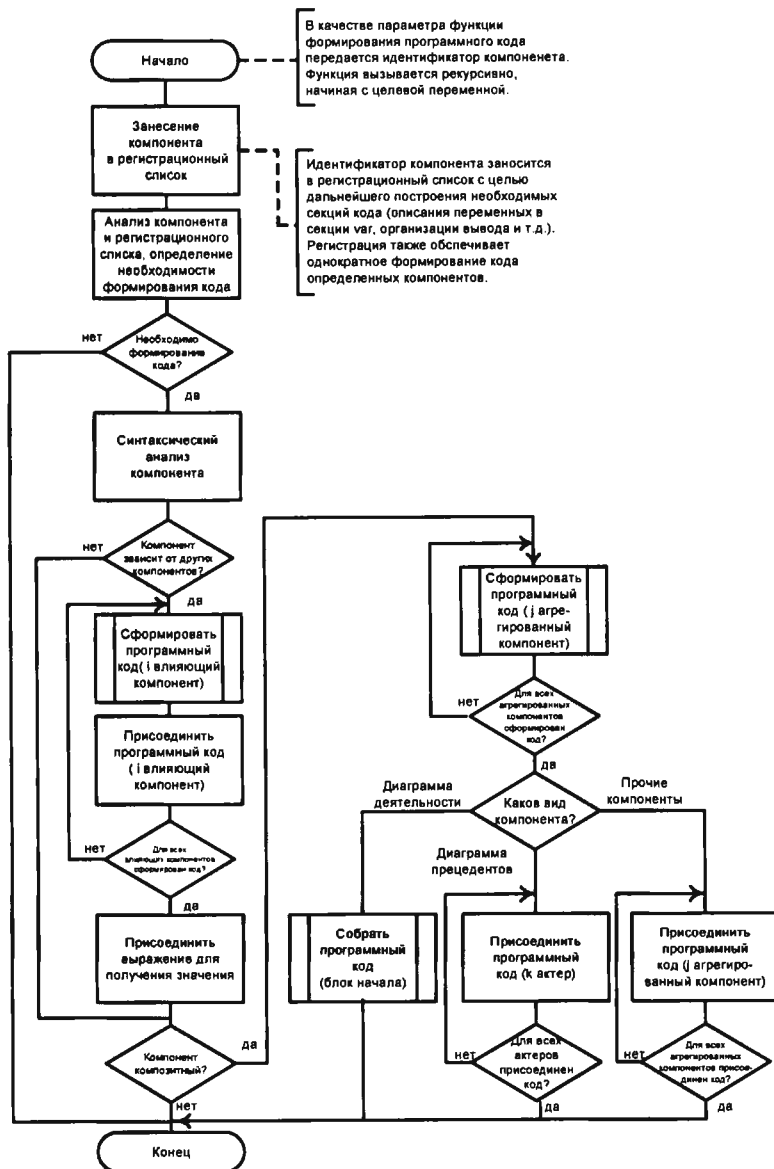
При построении программного кода по компоненту, зависящему от других, необходимо сформировать и присоединить программный код для всех компонентов, от которых зависит данный, и добавить необходимый код для расчета значения.

На рис. 13 представлен обобщенный алгоритм формирования программного кода по компоненту.

Разработан комплекс архитектурных моделей, позволяющий обеспечить программную реализацию предложенной концепции, метамодели, метода и алгоритмов в виде системы имитационного моделирования. Привлечение для этого объектно-ориентированного подхода обусловлено наличием разнообразных модельных компонентов, которые, тем не менее, имеют некоторые общие свойства и предполагают сходное поведение. Использование объектно-ориентированных методов, во-первых, позволяет учесть сложность проектируемой системы, во-вторых, предоставляет возможность сократить затраты труда за счет использования механизмов наследования и полиморфизма, в-третьих, обеспечивает гибкость и расширяемость программного решения.

Функционирование системы обеспечивается взаимодействием ее ключевых модулей. На рис. 14 представлена предлагаемая архитектура в виде диаграммы пакетов.

Структурные элементы системы могут быть разделены на несколько уровней или слоев. Три пакета верхнего уровня предназначены для организации взаимодействия системы с пользователем. Пакет генерации программного кода содержит классы, отвечающие за построение программного кода имитационной модели. Классы, соответствующие элементам метамодели, относятся к пакету переменных, а также к модулю блоков и диаграмм. Для сокращения степени зависимости между различными частями системы используется пакет, содержащий абстракции нижнего уровня.



Источник: Хубаев Г.Н., Широкова С.Н., Щербаков С.М. Автоматизированный синтез имитационных моделей деловых процессов // Известия вузов. Северо-кавказский регион. Технические науки. – 2008. – №4. – С. 73-79.

Рис. 13. Обобщенный алгоритм формирования программного кода имитационной модели по компоненту метамодели

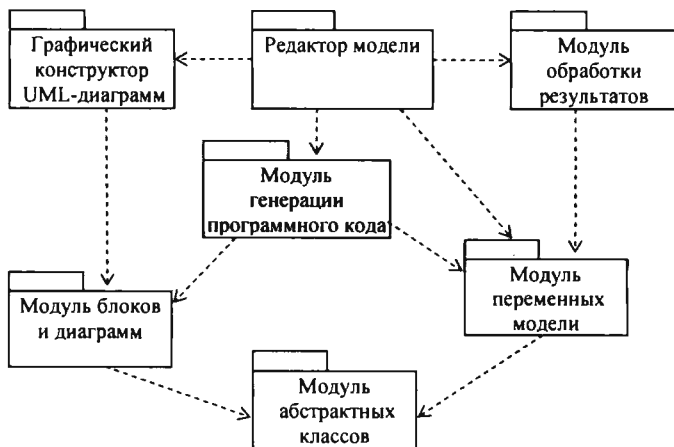


Рис. 14. Структура системы имитационного моделирования

Реализация алгоритмов формирования программного кода имитационной модели осуществляется путем взаимодействия различных программных объектов, которое может быть проиллюстрировано диаграммой последовательности, представленной на рис. 15.

Разработанная на основе предложенных архитектурных моделей программная система СИМ-UML⁵ предназначена для решения следующих задач: ведение перечня переменных имитационной модели, позволяющих задавать количественные характеристики изучаемого процесса; построение UML-диаграмм, определяющих структурные и поведенческие характеристики делового процесса; автоматическое формирование программного кода имитационной модели.

Система автоматизированного синтеза имитационных моделей реализует функции, необходимые для моделирования процессов использования интернет-приложений. Среди функций можно выделить:

- построение перечня переменных модели. Система имеет пользовательский интерфейс, позволяющий создавать переменные модели и задавать основные параметры для их имитационного моделирования: закон распределения и его параметры, формула расчета значения и т.д. в зависимости от вида переменной;
- моделирование переменных, заданных таблично;
- поддержка модельного времени и расчет интегральных значений;
- создание UML-моделей в графическом конструкторе системы. Встроенный графический конструктор позволяет построить диаграммы, связать их между собой и определить их количественные параметры;
- поддержка механизма дорожек. Позволяет определить исполнителей операций делового процесса и отследить затраты труда по исполнителям;

⁵ Хубаев Г.Н., Щербakov С.М., Рванцов Ю.А. Система автоматизированного синтеза имитационных моделей на основе языка UML «СИМ-UML» // Свидетельство об официальной регистрации программы для ЭВМ. - №2008615423. - М.: РОСПАТЕНТ, 2009.

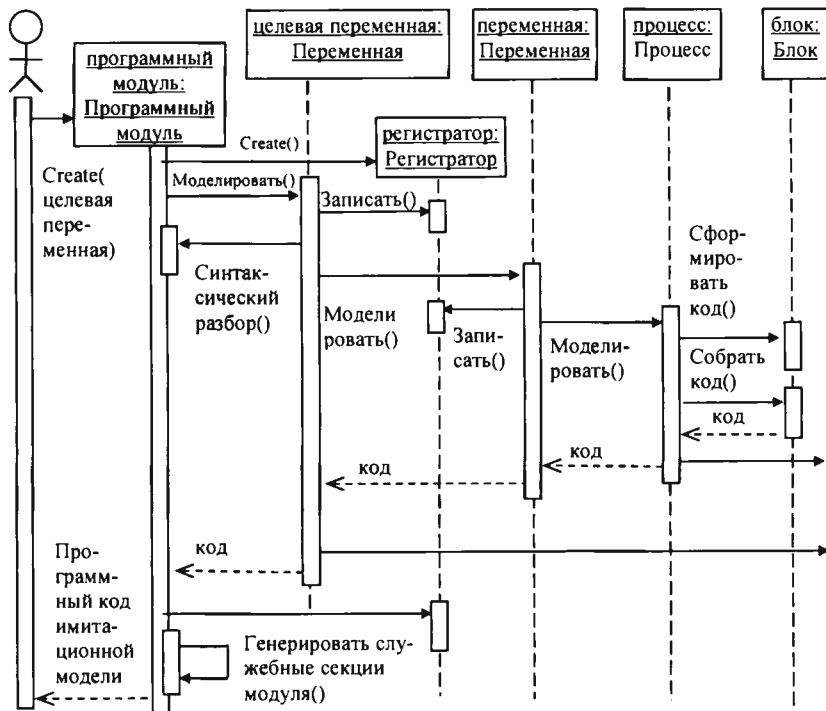


Рис. 15. Автоматическое построение программного кода имитационной модели

- проверка корректности диаграмм. Проверяется соответствие построенной диаграммы правилам языка UML. В частности, рассматривается допустимость построенных параллельных и альтернативных ветвлений, отсутствие закливания, корректность определения количественных параметров и т.д.;

- автоматическая генерация программного кода модели. На основе заданных визуальных и количественных компонентов система автоматически формирует программный код имитационной модели на языке Pascal. Подобная организация обеспечивает гибкость системы - созданные имитационные программные модули могут использоваться автономно или встраиваться в прикладную программную систему;

- создание консольных или снабженных графическим пользовательским интерфейсом программ для имитационного моделирования. Такие имитационные программы могут использоваться как вместе с системой имитационного моделирования, так и автономно;

- планирование имитационного эксперимента. Система позволяет задавать факторы имитационного эксперимента, определять их уровни и проводить имитационный эксперимент. Так, можно исследовать поведение изучаемой системы при изменении ее параметров и найти оптимальное сочетание значений факторов;

- получение основных статистических характеристик выходного параметра. В ходе имитационного моделирования рассчитывается множество значений выходных параметров модели - для каждой итерации моделирования. Состав переменных, включаемый в список выходных параметров, формируется пользователем. По полученным значениям рассчитываются основные статистические характеристики: среднее значение, дисперсия, коэффициент вариации, минимальное и максимальное значения, асимметрия, эксцесс, мода и др.;

- построение гистограмм значений выходных параметров моделирования формы закона распределения.

На рис. 16 показан в общем виде порядок работы с системой СИМ-UML.

Метод автоматизированного синтеза имитационной модели и система СИМ-UML позволяют проводить автоматизированное формирование имитационных моделей процессов функционирования интернет-приложений на основе их визуальных моделей.

В пятой главе «Визуальное и имитационное моделирование процессов эксплуатации интернет-приложений» рассматриваются вопросы использования инструментария интеграции визуального и имитационного моделирования для анализа и моделирования процессов эксплуатации интернет-приложений в деятельности организаций.

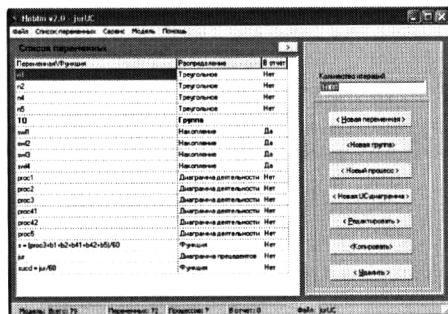
Модель может рассматривать один из аспектов интернет-приложения: сущности предметной области (концептуальная модель); цели и задачи; бизнес-процессы; сервисы; страницы и связи (навигационная модель); пользователи и права доступа; технологии и компоненты и т.д.⁶. Моделирование может использоваться: для принятия решений о построении или развитии интернет-приложения; для выбора одного из альтернативных вариантов построения интернет-приложения; для документирования системы; для коммуникации между разработчиками; для автоматизации разработки системы за счет применения CASE-средств.

Интеграция визуального и имитационного моделирования процессов функционирования интернет-приложений позволяет оценить принимаемые решения на различных стадиях процесса построения интернет-приложений с точки зрения экономической эффективности.

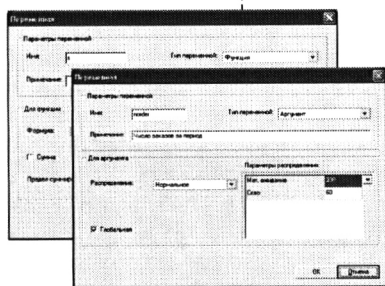
Предлагаемый подход к моделированию интернет-приложений подразумевает несколько принципов:

- моделирование затрат труда на построение и эксплуатацию интернет-приложений;
- использование визуальной модели интернет-приложения;
- итерационная модель жизненного цикла, с учетом особенностей интернет-приложения;
- интеграция визуального и имитационного моделирования и автоматизированный синтез имитационных моделей;
- моделирование деловых процессов, связанных с интернет-приложением.

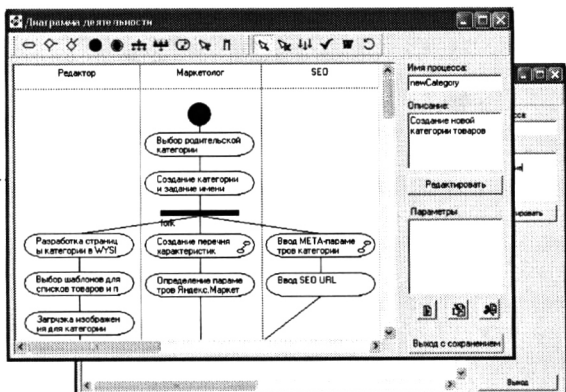
⁶ Web Engineering: Modelling and Implementing Web Applications / edited by Gustavo Rossi, Oscar Pastor, Daniel Schwabe and Luis Olsina. – Springer Verlag HCIS, 2007.



Создание переменных модели



Построение UML-диаграмм



Генерация программного кода имитационной модели

```
{uml}
function proc_proc5:real;
label L1, L7, L10, L17, L24,
L28;
var _s:real;
_nI,_nII:real;
sw14_1,sw13_1,sw12_1,sw11_1:real;
{usecase_forward}
{usecase}
begin
_s:=0;
sw14_1:=0;
sw13_1:=0;
sw12_1:=0;
sw11_1:=0;
```

Исполнение имитационной программы

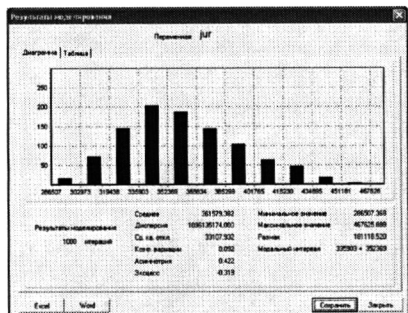


Рис. 16. Схема имитационного моделирования в системе SIM-UML

На рис. 17 показано использование модели интернет-приложения при каждой итерации процесса разработки или развития интернет-приложения.

На каждой итерации процесса развития интернет-приложения строится модель – отображения различных аспектов интернет-приложения. Решения, заложенные в модель, определяются поставленными целями модификации интернет-приложения и опытом эксплуатации его предыдущей версии. Изучение модели позволяет оценивать возможные решения, рассматривать различные варианты модификации и делать обоснованный выбор.



Рис. 17. Схема итерации процесса модификации интернет-приложения (с использованием имитационной модели)

Включение имитационного моделирования в рассматриваемую схему позволяет перейти на количественный уровень оценки интернет-приложения. Метод автоматизированного синтеза дает возможность построения имитационной модели интернет-приложения на основе UML-модели. Эксперименты с имитационной моделью позволяют: оценить предлагаемый вариант интернет-приложения; сравнить варианты возможных изменений; выработать проектные решения, отвечающие критерию экономической эффективности. Результаты имитационного моделирования закрепляются в модели интернет-приложения.

Предлагаемая методика анализа и моделирования интернет-приложений включает следующие шаги:

Шаг 1. Выделяются процессы эксплуатации интернет-приложения (пример UML-диаграммы процесса приведен на рис. 18).

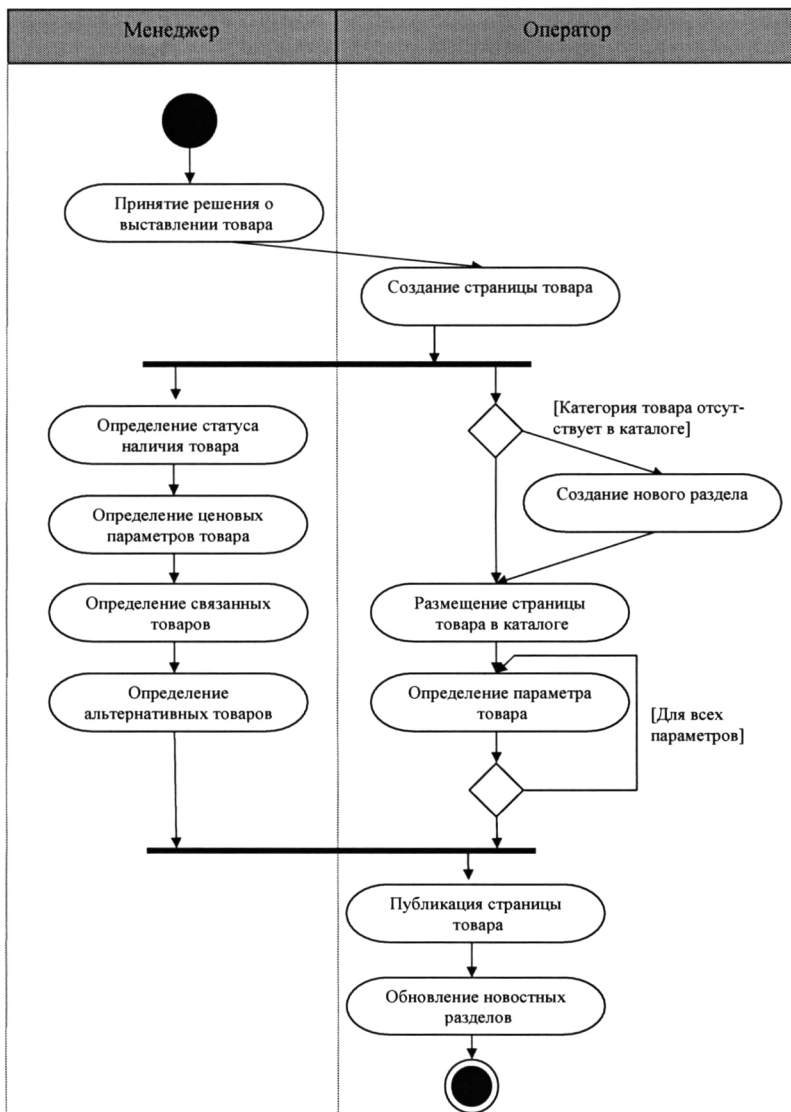


Рис. 18. UML-диаграмма процесса размещение товара в электронном каталоге

Деловой процесс включает последовательность отдельных операций, выполняемых определенными исполнителями: пользователями, администраторами и операторами интернет-приложения. Каждому варианту построения интернет-приложения будет соответствовать собственный набор деловых процессов.

Шаг 2. На основании статистических данных о деятельности организации или путем экспертного опроса определяются частотные параметры обращения к различным процессам. Некоторые обращения могут быть регламентными, для остальных должны быть заданы статистические характеристики частоты изменений (например, частота появления новой товарной позиции, изменения цены товара, приобретения товара и т.д.).

Шаг 3. Для каждой единичной операции путем хронометражных наблюдений или экспертным путем оценивается время ее выполнения (в случае привлечения экспертов они задают три значения - минимальное, максимальное и наиболее вероятное время выполнения операции).

Шаг 4. Процессы эксплуатации интернет-приложения представляются в виде UML-модели. Для описания структуры делового процесса работы с интернет-приложением используется диаграмма деятельности языка UML.

Диаграмма прецедентов языка UML позволяет описать моделируемую систему (интернет-приложение) в общем виде. При этом определяется перечень акторов (пользователей различных классов) и прецедентов (вариантов использования системы). Прецеденты связываются с деловыми процессами. Обращение актора к варианту использования инициирует исполнение соответствующего делового процесса (например, «Размещение товара в электронной витрине»).

Шаг 5. На основе UML-модели интернет-приложения автоматически формируется имитационная модель в соответствии с методом автоматизированного синтеза.

Шаг 6. Проводится имитационное моделирование. Результатом имитационного эксперимента являются статистические характеристики и законы распределения затрат труда для всех вариантов организации процессов эксплуатации интернет-приложения за выбранный период времени. Таким образом, оцениваются затраты труда каждого из пользователей, затраты труда по определенному подмножеству процессов и операций.

Имитационная модель позволяет проводить анализ влияния различных параметров интернет-приложения на величину затрат. Например, можно оценить, как изменится величина затрат труда на эксплуатацию интернет-приложения в случае заданного изменения объемов баз данных или интенсивности использования системы.

Интеграция визуального и имитационного моделирования процессов функционирования интернет-приложений позволяет определять затраты труда на исполнение процессов использования и развития интернет-приложений.

В заключении сформулированы выводы, основные положения и обобщения по результатам диссертационного исследования.

ПО ТЕМЕ ДИССЕРТАЦИИ ОПУБЛИКОВАНЫ

Статьи в изданиях из перечня ВАК РФ

1. Щербаков С.М. Анализ и моделирование интернет-приложений // Учет и статистика. – 2010. – №1 (27). – С. 68-75. – 0,5 п.л.
2. Щербаков С.М., Аручиди Н.А. Экономические аспекты построения интернет-приложений: методы сравнительного анализа и выбора интернет-технологий // Экономические науки. – 2008. – № 43. – С. 381-387. – 0,45 п.л. (лично автора – 0,3 п.л.).
3. Хубаев Г.Н., Щербаков С.М. Интеграция визуального и имитационного моделирования деловых процессов предприятия: принципы и инструментарий // Проблемы современной экономики. – 2008. – №3. – С. 252-258. – 0,5 п.л. (лично автора – 0,3 п.л.).
4. Щербаков С.М. Оценка экономической эффективности интернет-приложений на основе имитационного моделирования // Экономический вестник Ростовского государственного университета. – 2008. – №4. – Т.6. – С.128-131. – 0,4 п.л.
5. Хубаев Г.Н., Щербаков С.М., Широбокова С.Н. Язык UML как основа автоматизированного синтеза имитационных моделей // Вестник Ижевского государственного технического университета. – 2008. – № 4(40).– С. 181-185. – 0,35 п.л. (лично автора – 0,2 п.л.).
6. Широбокова С.Н., Щербаков С.М. Концепция и методика автоматизированного формирования имитационных моделей деловых процессов // Вестник Саратовского государственного технического университета.– 2008. – № 4(36). – С. 65-72. – 0,5 п.л. (лично автора – 0,3 п.л.).
7. Широбокова С.Н., Щербаков С.М. Метод и программная система имитационного моделирования на основе языка UML как инструмент анализа и моделирования деловых процессов // Вопросы современной науки и практики. Университет им. В.И. Вернадского. – 2008. – №4(14).– Т.2. – С. 154-161. – 0,4 п.л. (лично автора – 0,2 п.л.).
8. Щербаков С.М. Процессно-статистический подход в сфере государственного управления // Вопросы современной науки и практики. Университет им. В.И. Вернадского. – 2009. – №6(20).– С. 204-214. – 0,6 п.л.
9. Щербаков С.М. Визуальное и имитационное моделирование в жизненном цикле интернет-приложений // Известия КБНЦ РАН. – 2009. – №4(30).– С. 129-137. – 0,6 п.л.
10. Щербаков С.М. Имитационное моделирование экономических процессов на основе объектно-ориентированного подхода // Вестник РГЭУ «РИНХ». – 2009. – № 3(29).– С. 279-286. – 0,4 п.л.
11. Щербаков С.М. Имитационное моделирование экономических систем на основе языка UML // Аудит и финансовый анализ. – 2009. – № 6.– С. 447-455. – 1 п.л.

Монографии и научные издания

12. Щербаков С.М. Экономико-математическое моделирование интернет-приложений. Монография. – Ростов-н/Д.: РГЭУ (РИНХ), 2010. – 165 с. – 11 п.л.
13. Хубаев Г.Н., Щербаков С.М. Конструирование имитационных моделей в экономике и управлении. Монография. – Ростов-н/Д.: РГЭУ «РИНХ», 2009. – 176 с. – 11 п.л. (лично автора – 8 п.л.)
14. Джамурзаев Ю.Д., Стрельцова Е.Д., Хубаев Г.Н., Щербаков С.М. О некоторых подходах к оценке затрат труда и расчетам численности персонала (на примере налоговых органов): Научно-практическое пособие. – Ростов-н/Д.: РГЭУ «РИНХ», 2008. – 157 с. – 7,4 п.л. (лично автора – 2,5 п.л.).
15. Хубаев Г.Н., Щербаков С.М. Построение имитационных моделей для оценки трудоемкости деловых процессов с использованием языка UML: Препринт. – Ростов-н/Д.: РГЭУ «РИНХ», 2004. – 80 с. – 3,5 п.л. (лично автора – 2 п.л.).

Статьи в центральной и межвузовской печати

16. Щербаков С.М. Вопросы оценки экономической эффективности применения интернет-технологий // Учет и статистика. – 2005. – №3(7). – С. 199-205. – 0,5 п.л.
17. Хубаев Г.Н., Широбокова С.Н., Щербаков С.М. Автоматизированный синтез имитационных моделей деловых процессов // Известия вузов. Северо-кавказский регион. Технические науки. – 2008. – №4. – С. 73-79. – 0,6 п.л. (лично автора – 0,25 п.л.).
18. Щербаков С.М. Сеть INTERNET как инструмент получения статистических данных // Учет и статистика. – 2005. – № 1. – С. 40-49. – 0,75 п.л.
19. Саламатова М.А., Щербаков С.М. Моделирование деловых процессов в системе СИМ-UML (на примере торговой организации) // Системное управление. – 2009. – Выпуск 1(5). – [http://sisupr.mrsu.ru/2009-1/pdf/19_Salamatova.pdf]. – 0420900072/0019 – 0,75 п.л. (лично автора – 0,5 п.л.).
20. Суворов Д.В., Суворова А.Ю., Щербаков С.М. Система проведения экспертных исследований в сети Интернет: проектирование и разработка // Информационные системы, управление трудом и производством: ученые записки. Выпуск 10. – Ростов-н/Д.: РГЭУ «РИНХ», 2006. – С. 57-65. – 0,5 п.л. (лично автора – 0,2 п.л.).
21. Хубаев Г.Н., Щербаков С.М. Синтез имитационных моделей на основе диаграмм языка UML // Информационные системы, экономика, управление трудом и производством: Ученые записки. Выпуск 11. – Ростов-н/Д.: РГЭУ «РИНХ», 2007. – С. 108-121. – 0,7 п.л. (лично автора – 0,5 п.л.).
22. Щербаков С.М. Моделирование деловых процессов в системе СИМ-UML (на примере торговой организации) // Системное управление. – 2010. – Выпуск 2(8). – [http://sisupr.mrsu.ru/2010-2/pdf/scherbakov_1.pdf]. – 0421000072/0027 – 1 п.л.
23. Калугян К.Х., Щербаков С.М. Компьютерная система тестирования знаний как компонент информационной образовательной среды вуза // Вестник Академии. – 2005. – №1. – С. 61-66. – 0,5 п.л. (лично автора – 0,3 п.л.).

24. Лозина Е.Н., Щербаков С.М. Имитационное моделирование деятельности предприятия с использованием языка UML // Проблемы федеральной и региональной экономики: ученые записки. Выпуск 11. – Ростов-н/Д.: РГЭУ «РИНХ», 2008. – С. 166-171. – 0,33 п.л. (лично автора – 0,2 п.л.).

25. Хубаев Г.Н., Щербаков С.М. Особенности использования языка UML для имитационного моделирования // Проблемы федеральной и региональной экономики. Ученые записки. Выпуск 9. – Ростов-н/Д.: РГЭУ «РИНХ», 2006. – С.99-113. – 0,9 п.л. (лично автора – 0,6 п.л.).

26. Хубаев Г.Н., Щербаков С.М., Кудряшов Д.А. Определение численности сотрудников государственных учреждений на примере территориальных налоговых органов // Проблемы федеральной и региональной экономики: Ученые записки. – Ростов-н/Д.: РГЭУ «РИНХ», 2007. – с. 98-103. – 0,4 п.л. (лично автора – 0,2 п.л.).

27. Щербаков С.М. Модель оптимального выбора средства разработки Internet-приложения // Информационные системы, управление трудом и производством: Ученые записки. Выпуск 8. – Ростов н/Д.: РГЭУ «РИНХ», 2003. – С. 85-91. – 0,5 п.л.

28. Щербаков С.М. Основные направления исследования вопросов экономической эффективности интернет-приложений // Научный поиск: по страницам докторских диссертаций. Выпуск 6. – Ростов н/Д.: РГЭУ «РИНХ», 2007. – С. 125-132. – 0,4 п.л.

29. Щербаков С.М. Моделирование интернет-приложений электронной коммерции // Studium. – 2010. – № 2 (15). – [<http://www.sarki.ru/studium/publ10/sherbakov.pdf>]. – 04201000066/0017. – 0,6 п.л.

30. Джамурзаев Ю.Д., Щербаков С.М., Войнова Г.В. Имитационное моделирование трудоемкости выполнения функциональных операций работниками районных налоговых инспекций // Информационные системы, управление трудом и производством: ученые записки. Выпуск 9. – Ростов н/Д.: РГЭУ «РИНХ», 2005. – С. 30-42. – 0,5 п.л. (лично автора – 0,3 п.л.).

31. Щербаков С.М. Экономические аспекты процессов построения и использования интернет-приложений // Научный поиск: По страницам докторских диссертаций. Выпуск 7. – Ростов н/Д.: РГЭУ «РИНХ», 2008. – С. 97-104. – 0,5 п.л.

32. Щербаков С.М., Рванцов Ю.А. Имитационное моделирование деловых процессов на основе диаграмм языка UML // Проблемы федеральной и региональной экономики: ученые записки. Выпуск 11. – Ростов н/Д.: РГЭУ «РИНХ», 2008. – С. 171-179. – 0,45 п.л. (лично автора – 0,3 п.л.).

33. Аручиди Н.А., Щербаков С.М. Имитационное моделирование инвестиционных проектов в среде СИМ-UML // Studium. – 2009. – № 4 (13). – [<http://www.sarki.ru/studium/publ8/aruchidi.pdf>]. – 0420900066/0027. – 0,6 п.л. (лично автора – 0,4 п.л.).

34. Щербаков С.М. Принципы классификации интернет-приложений // Проблемы федеральной и региональной экономики: ученые записки. Выпуск 12. – Ростов н/Д.: РГЭУ «РИНХ», 2009. – С. 104-115. – 0,65 п.л.

35. Щербаков С.М. О статистическом подходе к изучению экономики сети Internet // Информационные системы, управление трудом и производством. Ученые записки. – Ростов-н/Д.: РГЭУ «РИНХ», 2003. – С. 16-18. – 0,25 п.л.

36. Щербаков С.М., Рванцов Ю.А. Система имитационного моделирования деловых процессов СИМ-UML // Информационные технологии моделирования и управления. – 2009. – №4(56). – С. 516-524. – 0,5 п.л. (лично автора – 0,3 п.л.).

37. Щербаков С.М. Моделирование интернет-приложений // Научный поиск: По страницам докторских диссертаций. Выпуск 8. – Ростов н/Д.: РГЭУ «РИНХ», 2009. – С. 75-86. – 0,7 п.л.

Статьи в Материалах Международных, Межгосударственных и Всероссийских конференций

38. Щербаков С.М. Экономические аспекты использования возможностей сети Интернет и интернет-технологий // Материалы XI Всероссийской объединенной конференции «Интернет и современное общество». – СПб.: СПбГУ, 2008. – С.121-124. – 0,25 п.л.

39. Широбокова С.Н., Щербаков С.М. Возможности метода и программного комплекса автоматизированного синтеза имитационных моделей деловых процессов // Компьютерное моделирование 2008: Труды междунар. науч.-техн. конф. – СПб: Изд-во Политехнического университета, 2008. – С.259-268. – 0,5 п.л. (лично автора – 0,3 п.л.).

40. Щербаков С.М. Применение экономико-статистических методов при проектировании Internet-приложений // Экономико-организационные проблемы проектирования и применения информационных систем: Материалы VI Всероссийской научно-практической конференции. – Ростов-н/Д.: РГЭУ «РИНХ», 2001. – С. 44-48. – 0,25 п.л.

41. Лозина Е.Н., Щербаков С.М. UML-модели как основа имитационного моделирования производственной деятельности // Труды VI Юбилейной международной научной конференции «Инновации в науке и образовании-2008». Часть 2.- Калининград: КГТУ, 2008.- С. 302-305. – 0,25 п.л. (лично автора – 0,2 п.л.).

42. Хубаев Г.Н., Широбокова С.Н., Щербаков С.М. Программный комплекс автоматизации синтеза имитационных моделей: принципы и применение // Моделирование. Теория, методы и средства: Материалы VIII междунар. науч.-практ. конф. – Новочеркасск: Лик, 2008.–Ч.1.– С.60-69. – 0,5 п.л. (лично автора – 0,2 п.л.).

43. Хубаев Г.Н., Щербаков С.М. Использование метода имитационного моделирования для оценки экономической эффективности системы информационной безопасности // Проблемы информационной безопасности. Материалы всероссийской научно-практической интернет-конференции. – Ростов-н/Д.: РГЭУ «РИНХ», 2007. – С. 27-33. – 0,25 п.л. (лично автора – 0,2 п.л.).

44. Щербаков С.М. Алгоритмическое обеспечение синтеза имитационных моделей деловых процессов // Экономические проблемы организации производственных систем и бизнес-процессов: материалы VII Междунар. науч.-практ. конф. – Новочеркасск: ЮРГТУ (НПИ). – 2009. – С. 13-20. – 0,45 п.л.

45. Щербаков С.М. Вопросы экономики сети Интернет // Технологии информационного общества - Интернет и современное общество: Труды IX Всероссийской объединенной конференции. – СПб.: СПбГУ, 2006. – С. 141-143. – 0,2 п.л.

46. Щербаков С.М. Сеть Интернет как экономическая система: основные свойства // Технологии информационного общества - Интернет и современное

общество: Труды VIII Всероссийской объединенной конференции. – СПб.: СПбГУ, 2005. – С. 87 - 88. – 0,2 п.л.

47. Щербаков С.М. Управление учебным процессом в условиях дистанционного образования с использованием Internet-технологий // Новые технологии в управлении, бизнесе и праве. Труды III международной конференции. – Невинномысск.: НИУБ и П, 2003. – С. 212-216. – 0,25 п.л.

48. Широбокова С.Н., Щербаков С.М. Автоматизация построения имитационных моделей деловых процессов: формализованная метамодель // Теория, методы проектирования, программно-техническая платформа корпоративных информационных систем: Материалы VI Междунар. науч.-практ. конф., г.Новочеркасск. – Новочеркасск: ЮРГТУ (НПИ), 2008. – С.125-133. – 0,45 п.л. (лично автора – 0,25 п.л.).

49. Щербаков С.М., Бойчук Н.А. Сравнительный анализ Internet - технологий по критерию функциональной полноты // Экономико-организационные проблемы проектирования и применения информационных систем: Материалы V Всероссийской научно-практической конференции. – Ростов-н/Д.: РГЭУ, 2000. – С.14-16. – 0,25 п.л. (лично автора – 0,2 п.л.).

50. Щербаков С.М. Моделирование и проектирование интернет-приложений: методы web-engineering // Компьютерные технологии в науке, производстве, социальных и экономических процессах (КТ 2009). Материалы X международной научно-практической конференции. – Новочеркасск: ЮРГТУ (НПИ). – 2009. – С. 28-39. – 0,5 п.л.

Зарегистрированные программные средства

51. Хубаев Г.Н., Щербаков С.М., Рванцов Ю.А. Система автоматизированного синтеза имитационных моделей на основе языка UML «СИМ-UML» // Свидетельство об официальной регистрации программы для ЭВМ. - №2008615423. – М.: РОСПАТЕНТ, 2009.

52. Хубаев Г.Н., Щербаков С.М., Кривошеева М.А., Голованов А.И., Победенный А.В. Программная система для оценки характеристик потребительского качества информационных продуктов для экономических приложений // Свидетельство об официальной регистрации программы для ЭВМ.- № 2004612668.- М.: РОСПАТЕНТ, 2004.

53. Хубаев Г.Н., Латыпов Р.Р., Щербаков С.М. Система имитационного моделирования «NOBLIN» // Свидетельство об государственной регистрации программы и включении в отраслевой фонд алгоритмов и программ.- № 50200100441.- М.: ГКЦ ИТ, 2001.

54. Хубаев Г.Н., Щербаков С.М., Аручиди Н.А. ПС анализа сложных систем по критерию функциональной полноты «Ireland» // Свидетельство об официальной регистрации программы для ЭВМ. - №2009615296. – М.: РОСПАТЕНТ, 2009.

55. Хубаев Г.Н., Щербаков С.М., Шибаев А.Л. Конструктор имитационных моделей деловых процессов // Свидетельство об официальной регистрации программы для ЭВМ.- № 2005612262.- М.: РОСПАТЕНТ, 2005.

102